Mandriva Enterprise Server 5.1 с. 13

LINUX FORMAT

Главное в мире Linux

Июнь 2010 № 6 (132)

ВАШ БЕСПЛАТНЫЙ DVD

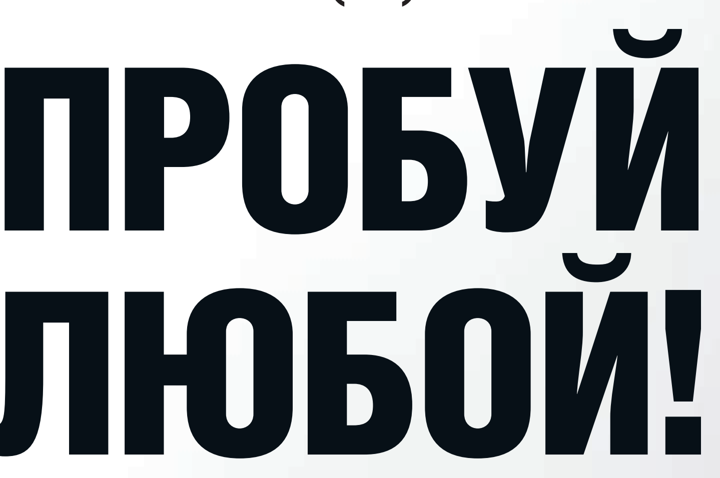
4 LINUX-СИСТЕМЫ

Новая жизнь с Unity, Dreamlinux, Sitaz и Absolute

А также: PC-BSD 8 » FlightGear » Open Clip Art

PC-BSD 8

Легкий путь в мир FreeBSD с. 9



ПРОБУЙ ЛЮБОЙ!

Сделайте шесть простых шагов к Linux-виртуализации, и тестируйте новые дистрибутивы, не рискуя старыми с. 22



32 страницы учебников на любой вкус!



Оснастим Firefox

Наша подборка лучших дополнений к web-браузеру с. 60

FlightGear 2.0

Садитесь за штурвал и запускайте ВСУ в свободном авиасимуляторе с. 11



Настоящая безопасность

Установите на ненужный ПК дистрибутив-брандмауэр и держите сеть под замком с. 16



«Когда другие уважают нашу работу, мы растем в их сознании как хорошие граждане»

Джоно Бэкон о правах человека с. 30



Плюс!

» ARM: чем он лучше Atom?

» CrossOver 9: Windows в Linux

» Zsh: оболочка для тех, кому мало Bash

» POV-Ray

Классический трассировщик лучей с. 64

» OOo Base

Создавать базы данных под силу каждому с. 72

» Google Calendar

Получайте уведомления о землетрясениях в виде SMS с. 76

Подписной индекс в каталоге агентства «Роспечать» 20882
Подписной индекс в каталоге «Пресса России» 87974
Подписной индекс в каталоге «Почта России» 16572





Что мы делаем

» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.

Кто мы

В этом месяце мы раскрываем для вас богатство возможностей оболочки *Zsh*, и решили спросить у команды: чего недостает *Bash*, де-факто стандарту в Linux?



Грэм Моррисон
Режима дежурных шуток. Включаете – и вместо команд автоматически подставляется **rm -rf /**.



Майк Сондерс
Управления на естественном языке, как в кино. Куда **cd** до «АКТИВИРОВАТЬ КАТАЛОГ КОД 4»!



Нейл Ботвик
Как фанат Gentoo, я люблю нечитаемые команды. **rh3kf@jt09+** вместо **echo** – пусть новички маются!



Эфраин Эрнандес-Мендоса
Функции управления одной только мышью: с кнопками, меню и разными красивыми картинками.



Эндрю Грегори
История команд – это круто, а как насчет предсказания команд на основе того, что я собираюсь сделать?



Энди Ченнел
Режима котэ. «Дорогой, ты сделал бэкап? Сожми файло, сожми файло <...>!». Да, я люблю демотиваторы.



Дэвид Картрайт
Тренажера клавиатуры, конечно. Будь он там, я нкиогда не деалл бы так монго очепяток.



Энди Хадсон
Расцветки приглашения в зависимости от опасности команды. **ls** – зеленым, **rm -rf** – красным, как файл в аду.



Ник Вейч
Помощника, вроде Скрепки в *MS Office*. «Похоже, вы собираетесь создать каталог. Вам нужна помощь?»



Сюзан Линтон
Command Command Revolution – вроде игры Dance Dance, но для технарей. **Cd ls, cd cd ls, df df rm**...



Шашанк Шарма
Команды **убейего**, удаляющей последний файл в каталоге и говорящей: «Ты выиграл!»



Маянк Шарма
Невинного режима, в котором мой терминал выглядел бы безобидно, пока я ломаю Пентагон.



Виртуальная реальность

По моим скромным подсчетам, на момент написания этих строк на DistroWatch было зарегистрировано 311 дистрибутивов. Может быть, их чуть больше (или чуть меньше), но общая картина налицо: разновидностей Linux в природе много. Это отчасти объясняет, почему свобода выбора применительно к свободному ПО периодически превращается в проблему выбора, и почему Ubuntu с его подходом «мы все решим за вас – занимайтесь своим делом» пришелся по душе столь многим.

Разумеется, «многим» не значит «всем» – и в мире есть немало людей, предпочитающих сделать свой выбор самостоятельно, даже если речь идет о столь прозаической вещи, как операционная система. Как быть им – ведь даже если отбросить из этих трехсот с лишним специализированные и региональные дистрибутивы, а также клоны Ubuntu, отличающиеся лишь экранной заставкой, число кандидатов все равно остается внушительным?

Лет десять назад мы порекомендовали бы вам выделить на диске специальный тестовый раздел (и периодически иметь проблемы с затертым загрузчиком). Лет пять назад – пользоваться LiveCD и мириться с малой скоростью работы (а также автоматически потерять системы, распространяемые только в форме установочных образов). Сегодня же мы посоветуем взять виртуальную машину – и знакомиться фактически с чем угодно, без риска, без существенных ограничений по скорости и почти что на настоящем оборудовании.

Конечно, виртуализация – тоже не панацея. Она едва ли поможет, если вы применяете Linux в основном для ресурсоемких задач, вроде 3D-моделирования или обработки видео, и вряд ли даст ответ на вопрос: «А какой дистрибутив лучше сработается с моим новым ноутбуком?». Подойдет ли она вам? У меня есть только один ответ: пробуйте, благо здесь вариантов уже не так много.

Валентин Синицын, главный редактор
» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Вопросы распространения: sales@linuxformat.ru
Web-сайт: www.linuxformat.ru
› Адрес редакции: Россия, Санкт-Петербург, Лиговский пр., 50, корп. 15
› Телефон редакции: (812) 309-06-86. Дополнительная информация на с. 112



Содержание

Ваш путеводитель по достопримечательностям июньского номера **LXF**



Обзоры

› **Книга: на сей раз всего одна, зато про Drupal.**

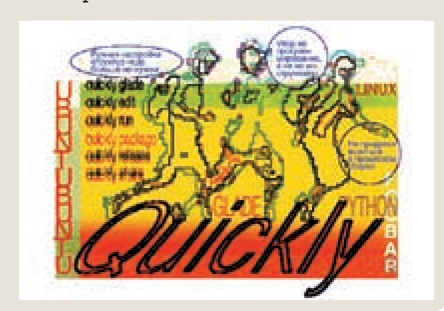


ПРОБУЙ ЛЮБОЙ!

Грэм Моррисон нацепил виртуальные наушники, чтобы познакомиться с новейшими релизами дистрибутивов без вреда для своей непорочной установки Fedora.

Что за штука...

Quickly



Где они водятся, зачем они нужны и с чем их едят?



Люди говорят



«Сначала моя реакция была той же, что выказали некоторые люди.»

Дистро-революция!

Четыре Linux и один Unix, а также:

LXFDVD132

» **OpenClipArt** Свободные изображения, чтобы сделать жизнь ярче
» **Brain Party** Лучшая Linux-игра от редактора британского **LXF**
» **FlightGear** Еще одна лучшая Linux-игра
...плюс интернет-шлюзы и кое-что еще! **с. 100**

Ищите в этом номере...

Постоянные рубрики



› **А вы не забыли оформить подписку на LXF?**

Учебники





› **Если вам вдруг потребуются картинки, их есть на нашем DVD.**

Кадры решают всё

Главной сложностью внедрения СПО в образовании является отсутствие подготовленных специалистов. Сравним подходы к решению этой проблемы, предлагаемые лидирующими проектами.



» Рубрику ведет ЕВГЕНИЙ КРЕСТНИКОВ

Начнем с компании «Альт Линукс», совместно с ФГУ ГНИИ ИТТ «Информика» запустившей недавно программу бесплатной сертификации. Получить сертификат с присвоением статуса «Специалист по внедрению ПСПО в системе образования» может каждый, кто помог образовательному учреждению внедрить продукты «Альт Линукс». По словам представителей компании, масштаб проекта не важен – соискателю достаточно отправить заявку и пройти онлайн-собеседование через *Jabber*. При этом «липовых» сертификатов выдавать не будут: «Если человек успешно провел внедрение, мы запрашиваем контактные данные этой фирмы и уточняем, действительно ли все было так, как он описал», говорит Анна Шадеева, руководитель отдела маркетинга «Альт Линукс». По словам г-жи Шадеевой, «сертификат юридического статуса не имеет. Он имеет отношение к продуктам «Альт Линукс»». На сайте компании тем временем заявлено, что он дает документированное подтверждение компетенции в ПСПО. Это не диплом государственного образца, но заверенный печатью предприятия и подписью руководителя документ, имеющий формальный статус любого другого фирменного сертификата. Главный маркетолог «Альт Линукс» с нами не согласилась: «Как может сертификат иметь юридическую силу, если ни в каких юридических документах не прописаны требования к кандидату, получающему данный статус?» – спросила она. Что на это ответить? В отличие от очевидных маркетинговых резонов, практический смысл данной программы неясен – можно выдать много бумаг с красивыми печатями, но проблему кадров это не решит.

Иной подход исповедуют в Mandriva.ru, предлагающей продукты с коммерческой поддержкой, сходные с «Альт Линукс 5.0 Школьный», на базе наработок проекта EduMandriva. Издательство БИНОМ и компания Mandriva.ru совместно разработали курсы MDV и курс «Основы преподавания GNU/Linux и свободного ПО», доступные для ознакомления на сайте etraining.mandriva.ru. «Альт Линукс» также занимаются через своих партнеров обучением, но в основном – подготовкой пользователей и технических специалистов.



» Курсы по применению открытых технологий в образовательном процессе можно найти в Сети.

В продвижении продуктов EduMandriva делает ставку на сотрудничество с учреждениями образования и создание на их базе центров компетенции. При этом «корпоративный» сегмент берет на себя Mandriva.ru: «Стратегия работы EduMandriva – работа с ИПК Про, вузами и другими учебными заведениями, работа через Mandriva.ru с партнерами по внедрению, работа с производителями ПО (сейчас с 1С тестируем новую «1С Школа» под Linux), работа в рамках единого репозитория (spohelp.ru)», говорит Александр Казанцев, руководитель EduMandriva. Что касается сертификации – по словам г-на Казанцева, подобные программы разрабатываются совместно с Центрами компетенции, издательством БИНОМ, ИПК Про и «ГНУ/Линуксцентром», и будут доступны к следующему учебному году.

Разница подходов очевидна: «Альт Линукс» действует по обычной коммерческой схеме, позволяющей зарабатывать на СПО. Компании интересно выигрывать тендеры, а также продавать учебным заведениям услуги технической поддержки. Вкладываться в решение кадровой проблемы «Альт Линукс» не может (на этом много не заработать), и ограничивается программой бесплатной сертификации. Mandriva.ru тоже хочет денег и тоже не может себе позволить серьезных инвестиций без отдачи в обозримом будущем. Но здесь лучше поняли специфику отрасли, и грамотно применили старую как мир схему «компания/сообщество», создав проект EduMandriva. Чья стратегия вернее, покажет время.

Новости короткой строкой

» В сети появилась информация о возможном приобретении Mandriva S. A. фирмой Linagora.

» Canonical представила новый мобильный интерфейс *Unity* и основанную на нем Ubuntu Light – урезанную версию системы, предназначенную для быстрого запуска.

» Запрет запуска альтернативных ОС на PS 3 стал предметом коллективного иска к Sony.

» SourceForge.net представила Downloads – сервис распространения программ без регистрации полноценного проекта.

» Минсвязи РФ планирует разработать российский аналог Google Docs.

Linux: [Пере]загрузка

В Red Hat разрабатывают замену классическому способу инициализации системы.

Леннарт Петтеринг [Lennart Poettering] представил на суд общественности новый механизм инициализации системы – *systemd*. В разработке проекта, цель которого – объединить достоинства «классики» с преимуществами современных подходов, приняли участие сотрудники Red Hat, Novell, IBM, Intel и Nokia. Предполагается, что в результате мы получим систему, совместимую с традиционными init-скриптами и командами, поддерживающую технологии *cgroups*, AutoFS и *D-Bus*. *Systemd* – не только система инициализации: она содержит функционал *cron*, *inetd*, менеджера графического входа в систему и даже (частично) *SELinux*.

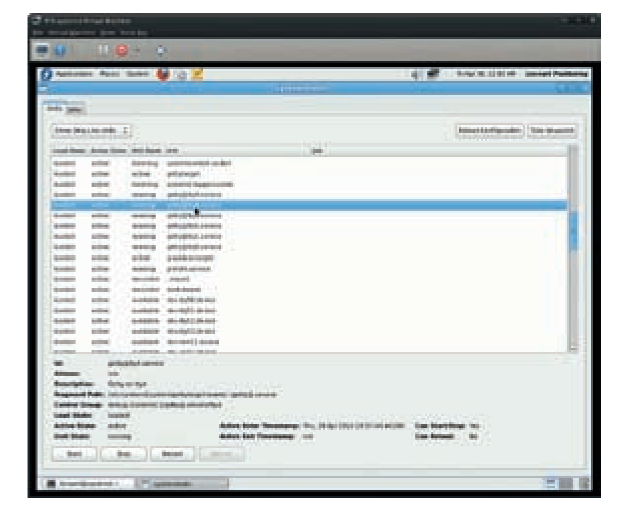
Здесь реализованы механизмы контроля сокетов, как в *launchd* из Mac OS X: система сама создает их, а затем запускает демоны. Кроме того, возможен автоматический запуск сервисов при обращении к сокетам. Сходный принцип используется для запуска служб, основанных на *D-Bus*.

Монтирование файловых систем (разумеется, исключая корневую, **/sys**, **/proc** и т.д.) происходит по требованию в фоновом режиме (используется механизм AutoFS). Леннарт также предлагает минимизировать число вспомогательных процессов (в основном это касается скриптов оболочки), переписав критичные участки на C и перенеся часть функционала в *systemd* и непосредственно в демоны (уже готовы подсистемы монтирования и установки имени хоста).

Еще *systemd* сможет отслеживать процессы и ограничивать их полномочия при помощи внедренного в ядро механизма контрольных групп (*cgroups*), а также осуществлять автоматический перезапуск демонов и многое другое.

Базовым элементом системы являются связанные между собой модули (**units**) различных типов, таких как **service**, **socket** или **device**. Вместо традиционных уровней загрузки появится их более гибкий аналог: модули **target** и **snapshot**, первый из которых представляет объединенную функционалом группу служб, а второй позволяет «запоминать» существующую конфигурацию модулей.

› *systemd* еще находится в стадии проработки концепции, но графический интерфейс для него уже есть.



От традиционных систем инициализации *systemd* отличается возможностью работы с зависимостями не только между службами, но и между событиями. А от другой «событийной» системы *upstart* – более высокой параллелизацией и расширенными возможностями контроля за процессами.

Поговорим о трендах

Мобильные Linux-платформы перестали быть «темой № 1»: в центре внимания сейчас оказываются «патентные войны».

В прошлом году мы уделяли много внимания мобильным Linux-платформам. Сейчас страсти вокруг этого направления поутихли, и вперед вырвалась другая тенденция – «патентные войны». В фокусе оказываются спецификации HTML5 и стандарты видеокодеков. В преддверии очередной «войны браузеров» в этом нет ничего удивительного.

Правда, первое событие, которое хотелось бы осветить, выбивается из общего потока. По сообщению агентства Reuters, в конце апреля корпорации Microsoft и HTC заключили соглашение о выплате лицензионных отчислений за используемые в «гуглофонах» HTC технологии MS. Представители Microsoft традиционно не говорят, какие именно патенты нарушены, но есть мнение, что речь идет о VFAT.

В этой связи следует упомянуть еще две интересные инициативы: Canonical получила у MPEG-LA лицензию на кодек AVC/H.264 (в этом можно убедиться здесь: http://www.mpegla.com/main/programs/AVC/Pages/Licensees.aspx), и поговаривают, что Nokia и Intel защитят платформу MeeGo от патентных рисков. Последняя новость основана на высказываниях некоторых представителей Nokia, а также на том факте, что разработчики *MeeGo* – слишком крупные игроки, чтобы игнорировать такие риски, и сами владеют немалым числом патентов.

Стоит также отметить недавний анонс *IE 9* (в новой версии ожидается поддержка тэга <video> и кодека H.264) и заявление менеджера проекта. Дин Хачамович [Dean Hachamovitch] сообщил, что Microsoft будет лоббировать в MPEG-LA продление сроков бесплатного использования H.264 для некоммерческой трансляции потокового видео. Стивен Джобс [Steven Jobs] также «засветился» в теме, причем на этот раз без связи с технологиями Adobe. В ответ на призыв FSFE включить в продукты Apple кодек Theora руководитель корпорации заявил, что считает Theora небезопасной в патентном плане, несмотря на открытость кодека.

А также...

» Мэтью Греган [Matthew Gregan] работает над аппаратно-ускоренным кодеком Theora для мобильных устройств.

» 1 мая суд признал несостоятельность претензий IP Innovation LLC к Red Hat и Novell. Речь шла о патенте Xerox/PARC на «пользовательский интерфейс с несколькими рабочими зонами».

Иллюстрация с сайта http://0pointer.de/blog/projects/systemd.html

Мобильные новости

Пусть эта тема и потеряла статус архиважной – нам все равно есть что по ней сказать.

Хотя такого накала страстей, как в прошлом году, сейчас нет, продвижение Linux на мобильных устройствах остается одной из важных тенденций. Как и ожидалось, пока лидирует Android – новые «гуглофоны» и «гуглопланшеты» анонсируются регулярно, а платформа активно развивается (количество приложений для Android, по неофициальным данным, превысило 50 000). Кроме того, вице-президент отдела разработок Энди Рубин [Andy Rubin] в своем интервью сказал, что в Android 2.2 (он уже должен выйти к тому моменту, когда вы будете читать эти строки) появится поддержка Flash.

Состоялся первый выпуск объединенной платформы *MeeGo* (**LXF131**). Для загрузки доступны образы системы для устройств на базе Intel Atom и смартфона Nokia N900. Текущая версия пока не включает графическую оболочку *MeeGo User Experiences*. Вероятно, Nokia и Intel решили объединить усилия, чтобы конкурировать с Android на всех фронтах. Думается, подобный шаг был очень своевременным – о поддержке *MeeGo* уже заявили ведущие производители. Правда, есть еще южнокорейская *Bada*, но, во-первых, она представляет собой очень гибкое решение, которое может быть основано не только на ядре Linux. А во-вторых, *Bada* нельзя назвать массовой, хотя в Samsung очень серьезно настроены сделать ее таковой.

> «В семействе мобильного Linux ждут интересного прибавления.»

Самой интересной «мобильной» сделкой кажется приобретение Palm корпорацией HP. Многих аналитиков смутила сумма: $1,2 млрд. ($5,70 за акцию) плюс еще $200 тыс. дополнительных платежей (в основном выплата долгов Palm). И все это – за компанию, единственными активами которой являются пылящееся на складе смартфоны и платформа webOS. Впрочем, аналитики признают, что фактическое банкротство Palm с качеством платформы не связано – новую webOS все наперебой хвалили. Шли разговоры, что она больше подходит для мобильных устройств, чем iPhone OS компании Apple. Вероятно, платформа и нужна HP – вскоре после сообщения о сделке в Сети поползли слухи о сворачивании работ над HP Slate и возможном появлении планшета на webOS. Кроме того, поговаривают об интересе HP к рынку смартфонов и коммуникаторов. Так что в семействе мобильного Linux ждут интересного прибавления.



› Linux – стандартная платформа для Nokia N900 – скоро появится и в других смартфонах.

Intel представила Moorestown

Программы – это прекрасно, но мобильному Linux нужен процессор, на котором они будут выполняться.

Корпорация Intel представила новую платформу для мобильных устройств, планшетных компьютеров и смартфонов, ранее известную под кодовым наименованием Moorestown. Она базируется на процессорах семейства Atom Z6xx и отличается от предыдущей линейки Menlow существенно (до 50 раз) меньшим энергопотреблением в режиме ожидания. Алексей Рогачков, ведущий специалист по внедрению продукции Intel в странах СНГ, рассказавший о новинке журналистам, подчеркнул, что мобильные гаджеты, в основе которых лежит новая платформа, смогут работать в режиме ожидания примерно 10 дней, 2 дня непрерывно воспроизводить аудиофайлы, 5–6 часов функционировать в режиме разговора и столько же – в режиме активного web-серфинга.

Новый процессор выполнен по 45-нанометровой технологии; его частота может составлять от 1,5 до 1,9 ГГц, на одном чипе с ним находятся графический контроллер, модуль кодирования и декодирования видео, а также контроллер памяти.

Поскольку новая платформа ориентирована на мобильные устройства, разработчики предусмотрели наличие всех коммуникационных интерфейсов, таких как Bluetooth, WiMAX, Wi-Fi, GPS и 3G. Кроме того, пользователи смогут наслаждаться видео в формате Full HD 1080 p.

«Для корпорации Intel новая платформа является во многом инновационным продуктом, – отметил Алексей Рогачков, – поскольку он отличается рекордно низким энергопотреблением, всего полватта, и крайне компактными размерами. Помимо этого, не стоит забывать, что он полноценно поддерживает все имеющиеся на сегодняшний момент интернет-технологии.

И, наконец, очень важным является наличие телефонного стека, которого раньше в процессорах x86 просто не существовало. Но для того, чтобы устройство успешно продавалось, необходимо предложить пользователям удобное программное обеспечение, оптимизированное под новую платформу».

В числе программных платформ, способных раскрыть весь потенциал нового творения инженеров корпорации Intel, были упомянуты Google Android, Moblin и MeeGo. Новые процессоры уже доступны для заказа производителям мобильных устройств, и ожидается, что первые продукты на их базе появятся на рынке до конца текущего года. LXF

LINUX FORMAT Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов



АЛЕКСЕЙ ФЕДОРЧУК
Его слабости – mass storage, разметка диска и файловые системы.

Fedora в новой сфере?

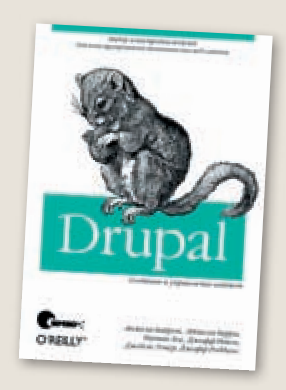
С первых дней своего отщепления от прародительского Red Hat'а дистрибутив Fedora рассматривался компанией как полигон для отработки новых технологий, и поэтому интенсивно обновлялся в межрелизный период. Что сугубо приветствовалось энтузиастами-экспериментаторами, но вызывало понятную настороженность со стороны «промышленного сектора начального уровня». В результате ниша «Red Hat для бедных» оказалась заполненной клонами, такими как CentOS и Scientific Linux.

Но похоже, что «командорской компании» это не понравилось. И отныне политика обновления Fedora меняется: в межрелизный период будут выпускаться только исправления безопасности; новые версий ядра и прочих основных компонентов системы отменяются. Что, конечно, пойдёт на пользу стабильности дистрибутива и откроет перед ним перспективы промышленного применения для тех, кто не нуждается в технической поддержке «настоящего» Red Hat или просто не может себе её позволить.

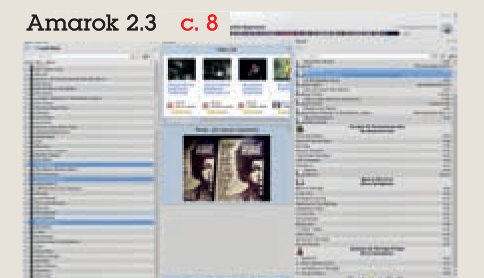
Но не потеряет ли Fedora своей фронтирной прелести для энтузиастов и экспериментаторов? Надеюсь, что нет. Думаю, что вместо двухступенчатой схемы дистрибуции – стабильного релиза и так называемого Rawhide сама собой сложится трёхступенчатая, подобная Debian'овской. Собственно, явочным порядком она и образовалась при подготовке 13-й версии. Каковую, надеюсь, заинтересованные лица смогут увидеть ко времени прочтения этих строк.

alv@posix.ru

Сегодня мы рассматриваем:

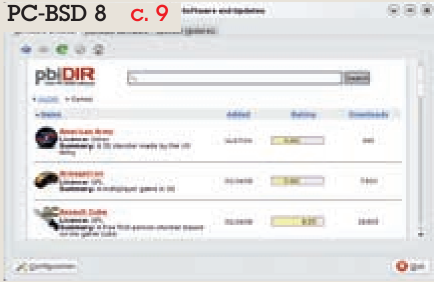


Amarok 2.3 с. 8



› Amarok сейчас лихо управляется с USB-носителями и обложками; настала пора обогнать iTunes?

PC-BSD 8 с. 9



› Устанавливается как Linux, выглядит как Linux и работает как Linux, но внутри это BSD.

Наш вердикт: Пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибалльной шкале (0 – низшая оценка, 10 – высшая). Как правило, учитываются функциональность, производительность, простота использования и цена, а для бесплатных программ – еще и документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.



Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но если разработчики рекомендуют *Autopackage*, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth

Разработчики: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Оправданность цены	9/10

» Если весь мир – сцена, то Google Earth – театр. Простая в использовании, захватывающая и ободряюще практичная программа.

Рейтинг 9/10

Amarok 2.3



Amarok пережил переход от популярной версии 1.4 к неудачной 2.0. **Нейл Ботвик** размышляет, вернет ли программе былую славу попытка 2.3.

Вкратце

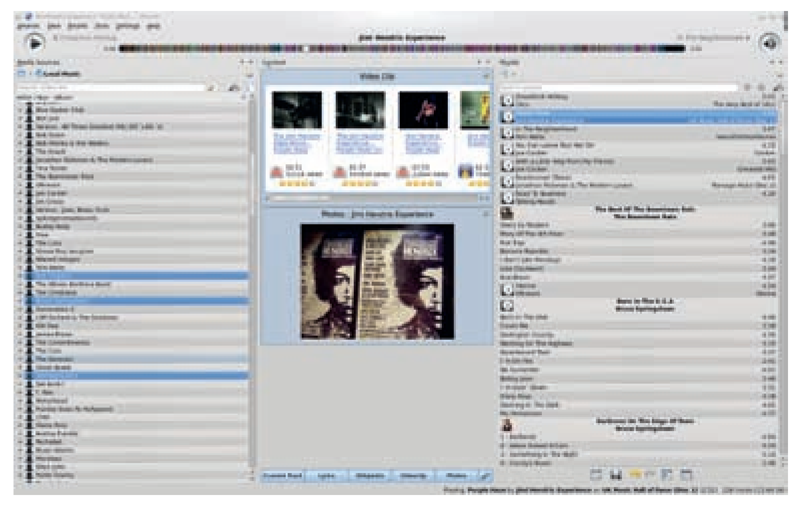
» **Функционально богатый аудиоплейер для воспроизведения записей локальной коллекции, CD, интернет-радиостанций и подкастов. См. также:** *Rhythmbox*.

Amarok 2 повторил судьбу KDE 4, которому во имя прогресса пришлось распрощаться с привычностью и стабильностью. Как и в случае с KDE 4, от каждой очередной версии ожидалось, что она все поправит. Перед нами *Amarok 2.3* – что же, поправил ли дело он?

Это, пожалуй, плейер не для тех, кто хочет просто воспроизводить музыку без лишних выкрутасов. Он для более требовательных слушателей, которым подавай загрузку подкастов и информацию о дорожках.

Amarok обладает всеми стандартными функциями музыкального плейера: скачиванием обложек, эквалайзерами и т. п. Есть и скриптовый интерфейс, так что функций можно еще и добавить, причем в ряде случаев – не выходя из программы. Например, можно устанавливать расширения для различных интернет-радиостанций и уведомлений о прослушиваемых композициях.

Базовый интерфейс программы состоит из панели инструментов с кнопками навигации и сведениями о проигрываемой дорожке, а также трёх областей: списка источников медиа, контекстной панели и плей-листа. Медиа-источником может быть музыкальная коллекция, служба типа Last.fm, файловая система, ваш собственный плей-лист или подписка на подкасты. На контекстной панели отображаются сведения о воспроизводимой дорожке. По умолчанию это параметры трека, слова песни или сведения об исполнителе из Википедии. Можно добавить автоматический онлайн-поиск видео- или фотоматериалов об исполнителе или композиции. Название третьей панели говорит само за себя: плей-лист и есть.



» **Стандартный интерфейс *Amarok* с тремя панелями: источник музыки, плей-лист и ссылки на видеоматериалы к текущей композиции.**

Это только базовая конфигурация – *Amarok* предоставляет пользователю полную свободу действий: панели можно масштабировать, закрывать, перетасовывать и даже перетаскивать за пределы главного окна. Перетаскивание одной панели поверх другой приводит к переключению на «вкладочный» вид – это удобно на устройствах со небольшими экранами.

Слишком сложно?

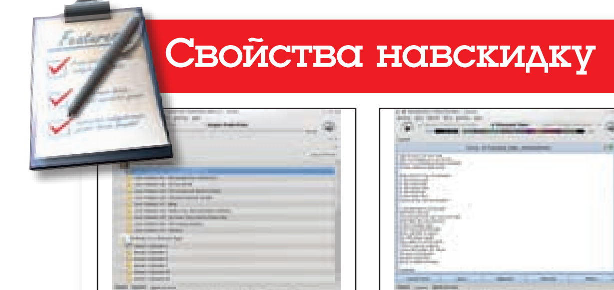
С подкастами программа обращается неплохо, хотя подписаться на них простым перетаскиванием ссылок не получится: придется открыть специальный мастер (Add Podcast). Новые эпизоды могут загружаться по мере появления, либо *Amarok* будет скачивать их по ходу воспроизведения. Возможно использование портативных плейеров как USB-накопителей, подключение к MTP или iPod с помощью *libgpod*. Приложение весьма сложное, а документации никакой. С базовыми функциями разобраться нетрудно, но если вы предоставляете кучу дополнений, гуманно было бы пояснить их применение.

Сведения о коллекциях сохраняются в базе данных *MySQL*. Это может быть встроенная БД с библиотеками *MySQL*, но *Amarok 2.3* несовместим с *MySQL 5.1*. Если у вас более поздняя версия, понадобится создать базу данных в *MySQL* и оформить в ней учетную запись для *Amarok*. В некоторых дистрибутивах этот процесс автоматизирован, но все же дружелюбия к пользователю здесь мало.

Воспроизведение записей не требует сверхнапряжения ни от процессора, ни от системы ввода-вывода, но *Amarok* имеет привычку сканировать музыкальные коллекции при старте. И если плейер запускается вместе с рабочим столом (а в случае с подкастами так, скорее всего, и будет), нагрузка на систему заметно возрастает.

В целом *Amarok* хорошая, мощная программа, хотя для многих слишком сложная. Основной удар по её популярности наносит нехватка документации, не позволяющая обычному пользователю добраться до самых интересных функций. LXF

Свойства навскидку



Подкасты

Подкасты можно слушать непосредственно из сети, либо *Amarok* будет загружать их автоматически по мере появления новых эпизодов.

Гибкий интерфейс

Окно можно реорганизовать согласно индивидуальным предпочтениям или размеру экрана.

LINUX FORMAT Вердикт

Amarok 2.3

Разработчик: Amarok
Сайт: http://amarok.kde.org
Цена: Бесплатно на условиях GPL

Функциональность	9/10
Производительность	6/10
Простота использования	5/10
Документация	3/10

» Продвинутый плейер с массой функций, но опорочен нехваткой документации для них.



Рейтинг 6/10

PC-BSD 8

Ценитель красоты графических инсталляторов **Коэн Верлоесем** открыл PC-BSD 8, Ubuntu мира BSD. Кто сказал, что BSD водится только на серверах?

Вкратце

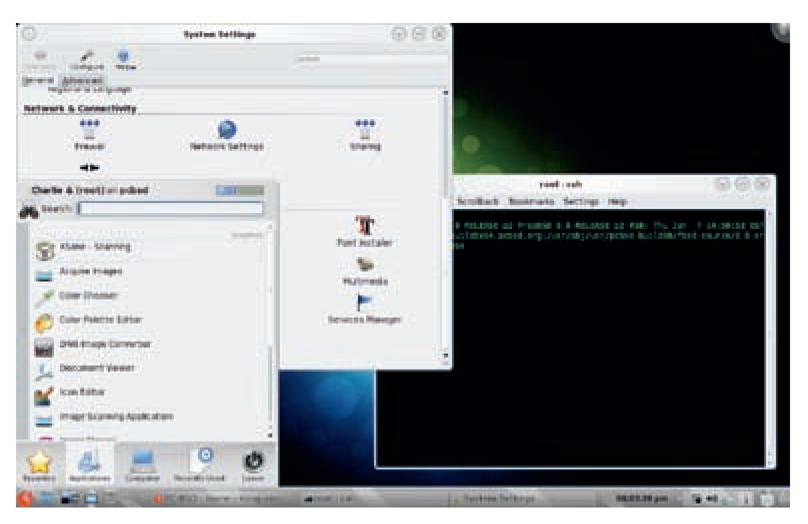
» Наипростейший способ развернуть полноценный рабочий стол KDE на FreeBSD 8. Если у вас есть время и тяга к познанию, попробуйте стандартный FreeBSD.

Мы часто поём дифирамбы BSD. Уж и стабилен он, и файловая система (как обоснованно полагают) много лучше, чем у Linux, и со стандартными приложениями работает... Единственное, чего не хватает BSD, так это дружелюбного инсталлятора. Точнее, графического инсталлятора нет вообще. Это серьёзный барьер, который не в состоянии преодолеть многие желающие познакомиться с интересной Unix-системой.

Дистрибутив PC-BSD заполняет пробел: его графический инсталлятор установит для вас рабочую среду KDE на системе FreeBSD 8. Поддерживается даже файловая система ZFS, которую стареющий *sysinstall* не тянет.

В некотором смысле, PC-BSD для FreeBSD – то же, что Ubuntu для Debian: созданный на базе основного дистрибутива пакет, доступный простому неопытному пользователю. Ubuntu – прекрасный способ познакомиться с миром Linux; PC-BSD – кратчайший путь в мир BSD. Судите сами: 20000 приложений, устанавливаемых на PC-BSD, без всяких изменений берутся из дерева портов FreeBSD. Различия настолько незначительны, что инсталлятор PC-BSD даже предлагает на выбор установку PC-BSD или FreeBSD 8, и вы получаете две операционных системы по цене одной – обе, правда, бесплатные. Ещё один бонус: инсталлятор предлагает попробовать Live-вариант, для проверки совместимости PC-BSD с вашим оборудованием перед установкой на диск.

> «На первый взгляд PC-BSD неотличим от дистрибутива Linux.»



› PC-BSD поставляется на 3,3-ГБ DVD или USB-брелке, и устанавливает модный рабочий стол KDE 4.3 на прочном фундаменте FreeBSD 8.

Заход в порты

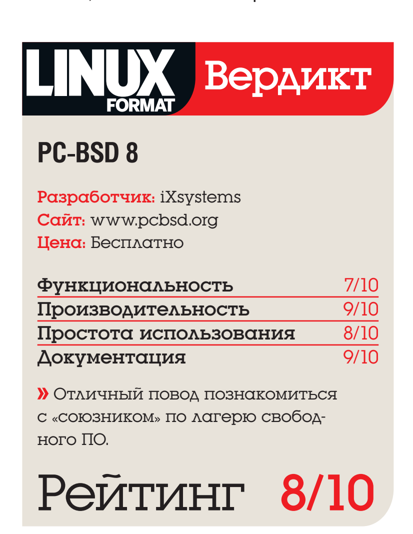
Хотя можно устанавливать приложения FreeBSD, компилируя их из коллекции «портов» (Ports; это «дальний родственник» пакетов SRPM в Red Hat и близкий – ABS в Arch и Portage в Gentoo), у PC-BSD есть собственная система управления готовыми двоичными пакетами, Push Button Installer (PBI). Пакеты – это файлы с расширениями **.pbi**; они устанавливаются по двойному щелчку.

Пакеты и их зависимости заносятся в отдельные автономные каталоги, во избежание конфликтов между версиями. Для установки и обновления PBI служит простая, но дружелюбная графическая программа *Software Manager*.

PC-BSD сосредоточен исключительно на KDE. Технически, можно установить и другую среду, например Gnome или *Xfce*, но официально такие варианты не поддерживаются, отчего вы потеряете в функциональности. Например, менеджер пакетов PC-BSD при установке приложений создаёт ссылки только для меню KDE.

На первый взгляд PC-BSD почти неотличим от дистрибутива Linux, но в нём явно не хватает ряда продвинутых функций. Например, поддержка 3D-режима все ещё в стадии эксперимента. Обижены владельцы видеокарт ATI: производитель не предоставляет свой проприетарный драйвер *fglrx* для PC-BSD или FreeBSD, и им придётся довольствоваться свободными драйверами. Если у вас видеокарта Nvidia, вам повезло больше: в PC-BSD предусмотрен проприетарный драйвер, дающий шансы на достойную поддержку трёхмерия. Правда, для 64-битных систем драйвер пока на уровне бета-тестирования (на нашей тестовой 64-битной системе проблем не было).

Рекомендуем попробовать эту новую систему пользователям KDE с технарской жилкой, склонным к экспериментам. LXF



LINUX FORMAT Вердикт

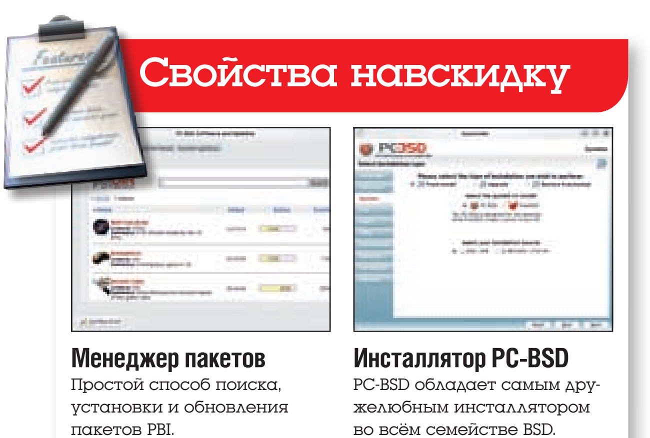
PC-BSD 8

Разработчик: iXsystems
Сайт: www.pcbsd.org
Цена: Бесплатно

Функциональность	7/10
Производительность	9/10
Простота использования	8/10
Документация	9/10

» Отличный повод познакомиться с «союзником» по лагерю свободного ПО.

Рейтинг 8/10



Свойства навскидку

Менеджер пакетов
Простой способ поиска, установки и обновления пакетов PBI.

Инсталлятор PC-BSD
PC-BSD обладает самым дружелюбным инсталлятором во всём семействе BSD.

CrossOver Pro 9.0

Мы постоянно охотимся за ПО, способным навек упразднить раздел C:\. **Энди Ченнел** считает, что CrossOver Pro 9 это по плечу.

Вкратце

» Среда для запуска Windows-приложений. См. также: *Wine* **(различные сборки),** *VMware* **и** *VirtualBox***.**

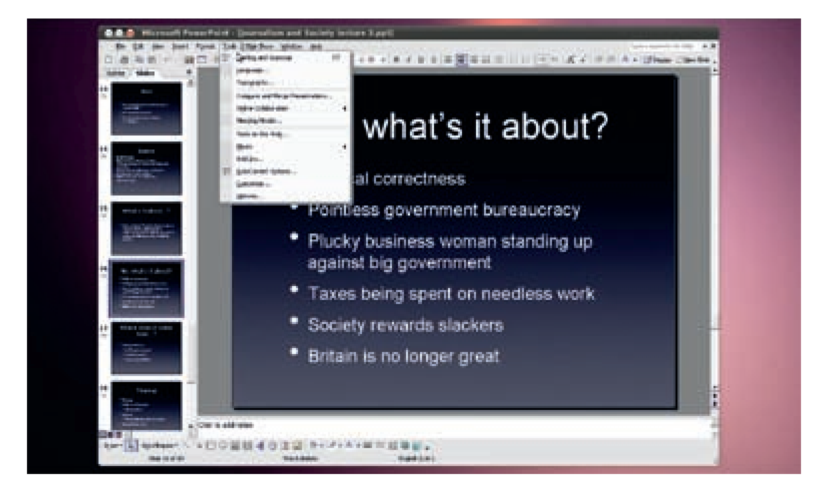
Всегда остаётся что-то одно. Маленькая программа, игра или утилита, из-за которой ну никак нельзя снести наконец раздел Windows. Пользователей-технарей и владельцев индивидуальных систем или небольших сетей выручает *Wine*. Но если требуется поддержка уровня предприятия и способность «раскатывать» инсталляцию в групповом режиме, то *CrossOver Professional 9.0* сбережет вам и силы, и время.

Поддержка приложений типа *Microsoft Office* и *Internet Explorer* улучшается с каждой новой версией, а настройка производительности позволяет некоторым программам (особенно *MS Office*) работать быстрее, чем «дома». Последний релиз замечателен полной переработкой интерфейса с использованием *GTK*, и теперь Windows-приложения больше похожи на Linux-собратьев, а кое-где появились даже «пульсирующие» окна.

> «Теперь Windows-приложения похожи на Linux-собратьев.»

Наше личное препятствие окончательному переходу – *Google SketchUp*, и мы с особым вниманием следили за усилиями *CrossOver* в этом направлении. Предыдущая версия вполне годилась для простой работы, однако более сложные проекты заметно «тормозили». Новый релиз еще не идеален, но *SketchUp* работает заметно быстрее и достаточно стабилен даже при большой нагрузке. Та же история со многими «неподдерживаемыми» приложениями, только вот

› Менеджер капсул-пакетов (*Bottle Package Manager*) создаёт RPM или Deb из работающих систем-«бутылок».



› ***CrossOver 9*** **запускает многие версии** ***MS Office*****, включая** ***2007*****. Это** ***Office XP*** **в имитации Win98 — отсюда и неровные шрифты.**

Adobe Creative Suite дальше CS2 продвинуть не удалось.

Полезное дополнение – вирусный сканер, который проверяет загружаемые файлы на чистоту и призван защитить бутылки-капсулы (экземпляры Windows) от инфекции.

Послание в бутылке

Трудности начинаются при нарушении планового порядка: интерфейс не выдает должную информацию о возникшей проблеме. Средство управления капсулами – и способность проникнуть внутрь псевдораздела **C:** каждой капсулы – способствует углублению понимания сути происходящего, но очень не помешала бы встроенная в *CrossOver* система уведомлений. С поддерживаемыми приложениями дело обстоит лучше. Установка быстрая, ПО работает как положено, а интеграция с рабочим столом дает знакомые ощущения.

Очень полезная функция – создание пакетов (RPM или Deb) из капсул. Это самодостаточный инсталлятор вместе со всеми установленными Windows-приложениями, который можно размножить на любое количество машин. Работает всё это безупречно: для проверки мы «пристраивали» весьма объёмистые капсулы в разные дистрибутивы, без каких-либо затруднений.

У нас возникли проблемы с настройками прокси в самом *CrossOver* – и при всем старании исправить их мы не смогли. Приложения, имеющие собственные средства работы с прокси, действуют нормально, но в системном масштабе возникают неприятности. Из-за прокси не удается установить некоторые загружаемые приложения – в частности, *IE7*. Можно предположить, что новый *CrossOver* берёт на себя системное управление прокси-функцией, но это не сработало ни на одной из наших машин. Мы составили сообщение об ошибке и нашли выход в установке параметров прокси из командной строки и последующем запуске *CrossOver* в том же сеансе терминала.

Это полезное, грамотное, экономически эффективное ПО. Но если вашего приложения нет в списке официально поддерживаемых, лучше проявить бдительность: прежде чем расставаться с деньгами, испытайте пробную версию. LXF



LINUX FORMAT Вердикт

CrossOver Pro 9.0

Разработчик: CodeWeavers
Сайт: www.codeweavers.com
Цена: $69,95 (версия Pro)

Функциональность	8/10
Производительность	6/10
Простота использования	7/10
Оправданность цены	8/10

» Неплохая альтернатива «родным» Windows-приложениям или виртуализации.

Рейтинг 7/10

FlightGear 2.0.0



Проштудировав пилотажное руководство «от доски до доски», **Алекс Кокс** наконец освоил самый передовой авиасимулятор открытого ПО...

Вкратце

» **Увлекательный, реалистичный авиасимулятор, способный покорить вашему пропеллеру весь мир. См. также:** *X-Plane*.

Центр управления полетами в ярости. Только что взлетел Cessna 172P Skyhawk, презрев радиоконтакт с диспетчером международного аэропорта Сан-Франциско. Заложив левый вираж, самолёт чиркнул колесами шасси по глади соседнего озера и, вопреки законам физики, отпружинил и снова взмыл к небесам. Неопытному авиатору явно недоставало навыков пилотирования.

Набрав высоту, лётчик решил, что с него хватит. Он двинул рукоять вперёд, и Cessna резко клюнула носом, рванувшись к расчерченной квадратами земле. Перегрузка бросила кровь к голове; поле зрения замутилось розовым. А потом был удар. Фюзеляж врезался в зелёное поле подобно копью, брошенному умелой рукой. Мир содрогнулся и в ужасе застыл.

Это, конечно, не подлинное событие. В реальной жизни бестолковость наказывается далеко не так зрелищно.



› **Путь обучения тернист — отвлекают глюки: *FlightGear* не для рассеянных, но упорство вознаграждает достойно.**

Это *FlightGear*, авиасимулятор с открытым исходным кодом, изощрённо реалистичный и нереально глючный.

Станем на крыло

Ваш первый опыт будет, вероятно, сходен с нашим. Муторная, но необходимая часть обучения лётному мастерству – освоение лёгкой, не в меру подвижной машинки. Дело в том, что Cessna – не только прекрасный учебный самолёт, но и единственное судно, по умолчанию снабжённое полным руководством. Причём «полным» в самом полном смысле. Оно проведет вас через предполётные проверки, радиопереговоры и стартовые процедуры. Вас не просто учат подниматься в воздух, а преподают все тонкости ремесла.

Отдадим должное многочисленным разработчикам *FlightGear*, посвятившим игре долгие часы своей жизни: поведение Cessna вполне реалистично. По крайней мере, оно такое же, как в имитаторах типа *X-Plane*, особенно при наличии джойстика (хотя он не обязателен).

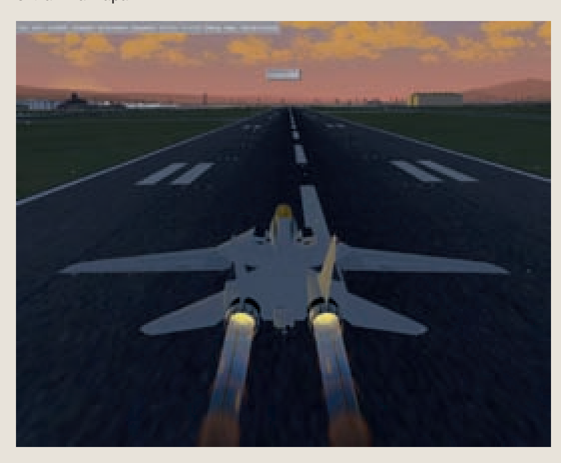
Неплохо бы улучшить отображение окружающей среды. Схематичные виды городов не радуют глаз – правда, это извиняется тем фактом, что карта охватывает весь мир. Версия 2.0.0 получила реалистичные объёмные облака, но они графически «привязаны» к самолёту: вы входите в пике, а облака устремляются вслед за вами. Взаимодействие с землёй недоотлажено, поэтому посадка превращается в серьёзное испытание.

Впрочем, полюбив игру, вы забудете о мелких неприятностях. *FlightGear* – бесспорный успех. Основательный подход делает игру весьма правдоподобной и познавательной. Да, поначалу будет трудненько, но и в этом есть доля привлекательности. Победная эйфория от собственноручного вывода в небо «Боинга-777» весьма ощутима, особенно после того, как вы одолели руководство по авионике, собственно «777» и серию учебников, чтоб подняться метра на три. LXF

Увидеть мир

Хотя исходный дистрибутив *FlightGear* охватывает лишь небольшую зону вокруг Сан-Франциско, к вашим услугам весь мир – стоит только захотеть. Для каждого участка земной поверхности воссоздан рельеф с точностью до площадок 10×10 градусов. Специальное приложение *Terrasync* следит за загрузкой нужных данных, когда ваша траектория выходит в новую зону.

Познавая мир, вполне можно повстречать других пользователей *FlightGear*. Интернациональный многопользовательский компонент игры не обязателен, но рекомендован – хотя бы потому, что соотечественники сумеют объяснить вам, как и что делать. Модуль расширения укажет вам через Google Maps, какие пилоты где летают – и уж тогда не уроните себя, выберите достойное судно из обширного онлайн-ангара.



› **Сайт *FlightGear* ломится от самолётов, вертолётов и даже автомобилей, созданных пользователями.**

LINUX FORMAT Вердикт

FlightGear 2.0.0

Разработчик: команда FlightGear
Сайт: www.flightgear.org
Цена: Бесплатно на условиях GPL

Функциональность	7/10
Производительность	6/10
Простота использования	9/10
Документация	9/10

» Небезупречная, но увлекательная игрушка для ветеранов аналогового джойстика.

Рейтинг 8/10

Bluefish 2.0.0



Гибкий текстовый редактор упрощает web-разработку, но **Боб Мосс** разочарован отсутствием предпросмотра.

Вкратце

» Текстовый редактор с поддержкой web- и прикладного программирования. См. также: *Quanta Plus* или *Kate*.

Web-разработка – своего рода запретная тема в сообществе Linux. У нас нет решений, подкрепленных мощным научно-исследовательским бюджетом, подобных коммерческим *Dreamweaver* или *Expression Web*. Однако многие из нас согласится, что *Bluefish* – лучшее из предлагаемого СПО в данной сфере (в Сравнении **LXF115** эта программа даже вышла в победители).

По сравнению с коммерческими пакетами *Bluefish* быстр и отзывчив, а системных ресурсов расходует где-то втрое меньше. Программа очень стабильна и может работать с тысячами документов одновременно.

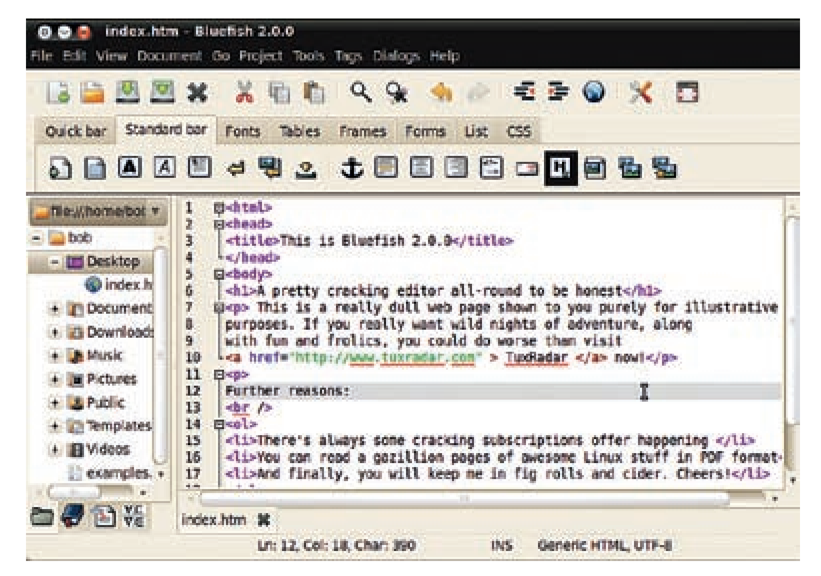
Всё, чего вы ждете от текстового редактора – проверка правописания, поиск и замена, многопользовательский режим, поддержка нескольких языков программирования, закладки для часто редактируемых файлов и одновременная работа над несколькими документами – здесь под рукой. Интерфейс интуитивно понятен и не загромождён; меню можно «разорвать», разместив их так, как требует рабочий процесс. Мощь программы не нервирует пользователя.

> «Подлинный конек Bluefish – это web-разработка.»

Но подлинный конёк *Bluefish* – именно web-разработка. Да, есть подсветка кода для C++ и других языков, но при работе с PHP, HTML, JavaScript и CSS вы получаете целую россыпь панелей и функций. Желаете увидеть в позиции курсора шаблонный код для встраиваемого CSS-стиля или вызвать мастер для создания формы, таблицы либо списка? Все это здесь есть – просто нажмите на соответствующую кнопку.

Гвоздь программы *Bluefish* – автозавершение HTML-кода. Пользователям *Dreamweaver* знакома автоподстановка (с автозакрытием) тэгов при кодировании – теперь *Bluefish* перенёс ее в Linux. Действует функция небезупречно, но время кодирования web-страниц сокращает.

Ещё одно достоинство *Bluefish* – богатая документация, особенно подробные вики-материалы и man-страницы.



» *Bluefish* – превосходное средство web-разработки, но отсутствие режима WYSIWYG в столь мощной программе вызывает недоумение.

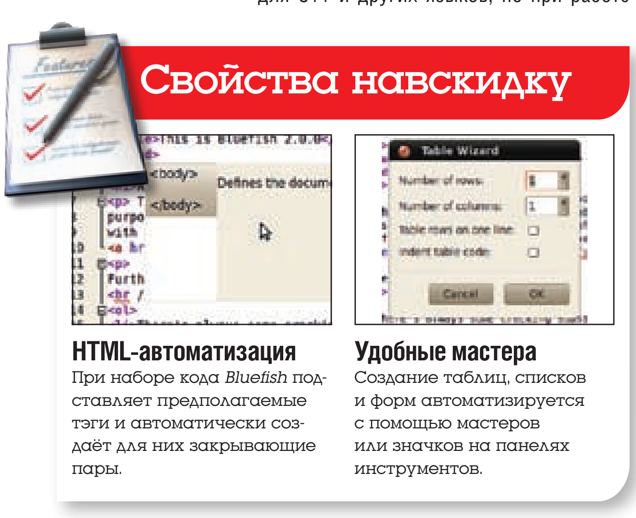
Зияющий пробел

Следует упомянуть и слона, которого поначалу не приметили: *Bluefish* не поддерживает WYSIWYG (визуальное редактирование). Возможность создавать web-страницы в этом режиме (или с предпросмотром результатов кодирования в отдельном окне) очень бы пригодилась.

В мире Linux немало редакторов с поддержкой WYSIWYG, и отсутствие таковой в *Bluefish* нас расстроило. Нет даже кнопки вызова браузера, в котором можно было бы открыть созданную страницу. Кроме того, при правке исходных текстов вы извлечете больше пользы из вашего IDE, чем из *Bluefish*. Здесь можно, конечно, вызывать внешние скрипты и утилиты (например, *make*), но встроенного компилятора и отладчика они не заменят.

Не поймите превратно: нам нравится мощь и гибкость *Bluefish*, но авторы, похоже, не пришли к выводу, делать ли им полноценное средство web-разработки или редактор для программистов, остановившись где-то посередине с «редактором за всё». Это и топит программу, а жаль: *Bluefish* мог бы стать единой средой web-разработки для всех Linux-пользователей. LXF

Свойства навскидку



HTML-автоматизация

При наборе кода *Bluefish* подставляет предполагаемые тэги и автоматически создаёт для них закрывающие пары.

Удобные мастера

Создание таблиц, списков и форм автоматизируется с помощью мастеров или значков на панелях инструментов.



LINUX FORMAT Вердикт

Bluefish 2.0.0

Разработчик: Bluefish Developers
Сайт: http://bluefish.openoffice.nl
Цена: Бесплатно на условиях GPL

Функциональность	7/10
Производительность	9/10
Простота использования	7/10
Документация	7/10

» Разносторонний текстовый редактор, который замахивается на всё и вся; но не лишен недочетов.

Рейтинг 7/10

Mandriva Enterprise Server 5.1

Решения Mandriva известны своим удобством, но оправдана ли цена? Разбирался **Евгений Крестников**.

Вкратце

» **Коммерческий серверный дистрибутив с поддержкой на русском языке. См. также: RHEL, CentOS или SLES.**

Ключевое ПО

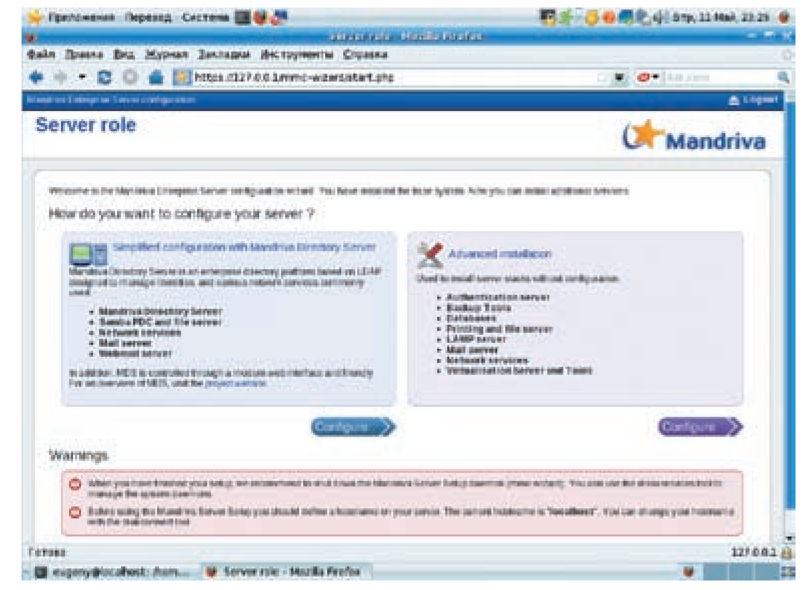
» Ядро 2.6.27
» *Xen* 3.3.1, *KVM*
» Mandriva Directory Server, OpenLDAP 2.4, *Kerberos* 5, *Samba* 3.2
» *Tomcat*, Sun JDK, JRE (1.5 и 1.6), OpenJDK
» *Postfix* 2.5, *Sendmail*, Cyrus-IMAP 2.3, Courier-IMAP 3.0
» *Apache* 2.2, PHP 5.2, Perl 5.10, *Squid* 3.0
» *MySQL* 5.0, *PostgreSQL* 8.3
» *Bacula* 3.0.1

Mandriva Enterprise Server (MES) 5.1 – корпоративный сервер, совместимый со спецификацией LSB (Linux Standard Base) 4.0 и отличающийся длительным сроком поддержки (5 лет). В его состав включено популярное ПО, а также собственные разработки Mandriva, упрощающие ввод в эксплуатацию. Подписчикам доступна техническая поддержка на русском языке.

Это коммерческий продукт. Чтобы получить доступ к двоичным обновлениям и техническую поддержку, необходимо оформить подписку (годовую или трехлетнюю). Она осуществляется специалистами Mandriva.ru по телефону и электронной почте. Время реакции зависит от критичности возникшей проблемы, а также типа подписки: Базовая, Стандартная, VIP – последняя предполагает трехчасовой отклик в режиме 24×7 на запросы с высоким приоритетом (ситуации, способные привести к неработоспособности или серьезным повреждениям систем клиента). Бесплатный период для тестирования продукта – 1 месяц.

Развернем и проверим

Инсталляция MES 5.1 не вызовет затруднений даже у начинающих администраторов. Стоит отметить, что если MES устанавливается в качестве второй ОС, лучше выбирать вариант ручного разбиения дисков – при автоматическом у нас возникли проблемы с некорректным созданием разделов. Инсталлятор прост и понятен; он позволяет указать необходимое ПО (в том числе на уровне пакетов) и настроить сервисы, включая межсетевой экран и профиль безопасности.



» **Подобно eBox Platform (см. стр. 18), MES можно администрировать через web-интерфейс. Правда, на английском языке.**

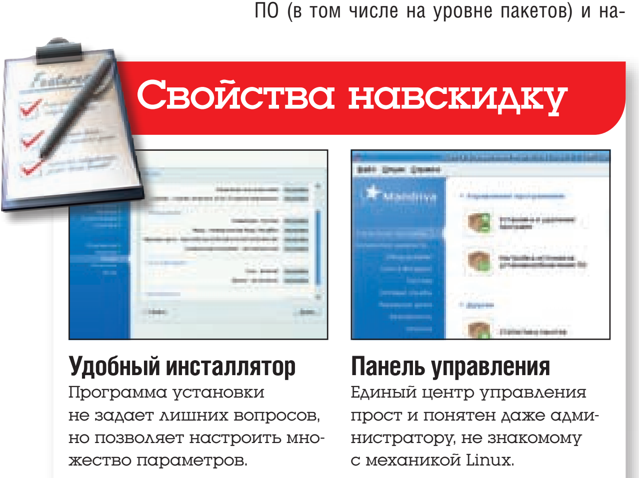
Настройка сервера осуществляется через единый *Центр управления* (стандартный для Mandriva). Он позволяет управлять пакетами и службами, администрировать пользователей и группы, работать с оборудованием и т.д. Также доступен web-интерфейс *MMC* (*Mandriva Management Console*): при инсталляции системы (если выбрана опция Mandriva Server Setup) ставится только пакет *mmc-wizard*, который позволяет произвести установку и первичную настройку различных служб.

Отдельного упоминания заслуживает *Mandriva Directory Server* – инструмент для ведения каталога пользователей, прав доступа к общим ресурсам локальной сети и управления сервисами. Он также настраивается через web-интерфейс, интегрируется с *Microsoft Active Directory* и способен заменить контроллер домена на Windows. Кроме того, подписчики MES 5.1 могут установить *Pulse II* – решение для централизованного управления ИТ-инфраструктурой компании.

Документация MES 5 показалась нам достаточно подробной, однако она доступна только на английском языке. Сама система, за исключением web-интерфейса *MMC*, локализована довольно неплохо.

Что в итоге?

Mandriva представила хороший продукт корпоративного уровня: MES 5.1 вполне способен заменить серверные ОС от Microsoft, а его развертывание не вызовет затруднений даже у новичков. Отсутствие русскоязычной документации компенсируется наличием поддержки на родном языке. Для использования в сетях малых и средних предприятий этот дистрибутив можно рекомендовать смело. Что касается масштабных внедрений – для начала, мы бы обратились в Mandriva.ru, чтобы изучить чужой опыт. LXF



Свойства навскидку

Удобный инсталлятор
Программа установки не задает лишних вопросов, но позволяет настроить множество параметров.

Панель управления
Единый центр управления прост и понятен даже администратору, не знакомому с механикой Linux.



LINUX FORMAT Вердикт

MES 5.1

Разработчик: Manriva S.A.
Сайт: www.mandriva.ru
Лицензия: Коммерческая
Цена: От 12500 рублей

Функциональность	9/10
Производительность	9/10
Простота использования	10/10
Документация	7/10

» Простой в использовании серверный дистрибутив, с русскоязычной технической поддержкой.

Рейтинг 9/10

Drupal: Создание и управление сайтом

Можно ли сделать кривую обучения замечательной CMS *Drupal* несколько более пологой? Разбирался **Александр Кузьменков**.

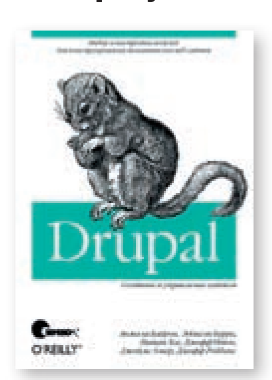
Система управления содержимым *Drupal* существует далеко не первый год, и уже обросла внушительной армией пользователей и разработчиков сторонних модулей. Как и у всех достаточно развитых и претендующих на универсальность продуктов, у *Drupal* есть довольно серьёзный недостаток: высокий порог вхождения, который со временем отнюдь не становится ниже.

Эта книга затевалась как решение данной проблемы. Авторы, кстати, принимают активное участие в разработке *Drupal*, и они в курсе того, чем живёт проект. Удалось ли им реализовать задуманное?

Получилась книга для практиков, готовых сразу, без лишних слов, ринуться в омут создания своего сайта. То есть типичный сборник рецептов. В первой главе даны краткий обзор CMS, рассказ об истории проекта и ответ на вопрос: «А зачем нам CMS вообще?». А уже во второй пойдёт сооружение из подручных модулей простенького сайта-визитки. В следующих главах авторы поступают аналогично: ставят задачу и тут же демонстрируют, как решить её с помощью некой комбинации готовых модулей. Очень порадовало, что среди описанных в книге модулей встречаются и сторонние, то есть авторы весьма добросовестно подошли к задаче – показать, как решить проблему, а не просто описать *Drupal* как вещь в себе. Приятно удивила и глава про интернационализацию – редкую птицу в работах англоязычных авторов.

Что до недостатков, то они типичны для любого сборника рецептов. Это недостаток теоретической базы: если вам потребуется быстро сделать нечто вне рамок рассмотренных в книге примеров, придётся сначала вникать в самые основы. Или искать решения в сети; кому что ближе.

Совершенным новичкам книга не подходит: нужно хотя бы в общих чертах представлять, что такое web-сайт изнутри и как он работает. Прилагаемое руководство по установке и обновлению *Drupal* не спасает. Профессионалам, знающим PHP, нечего рассчитывать увидеть здесь архитектуру системы. Зато нулям в программировании лучшее руководство для обучения через практику будет найти сложно. LXF



› **Книга — типичный сборник рецептов; понравится ли она вам, зависит от вашего отношения к данному жанру.**



Drupal: Создание и управление сайтом

Авторы: Анжела Байрон, Эддисон Берри, Натан Хог и др.
Издательство: Символ-Плюс/O'Reilly
ISBN: 978-5-93286-174-5
Цена: 590 руб.
Объем: 576 стр.

» Подойдёт ориентированным на результат практикам, запутавшимся во всех этих бесчисленных модулях.

Рейтинг 6/10

Книгу для данного обзора предоставлены интернет-магазином Books.Ru.

КОМПЛЕКСНОЕ РЕШЕНИЕ ДЛЯ МАЛОГО И СРЕДНЕГО БИЗНЕСА

На базе Windows*

MS WINDOWS SERVER
Сервер приложений
1С:Предприятие
MS SQL

MS WINDOWS SERVER
Терминальный сервер
Почтовый сервер
MS Exchange

MS WINDOWS SERVER
Файловый сервер

MS ACTIVE DIRECTORY
Контроллер домена

Рабочие станции с Windows

Телефонная станция и IP-телефоны

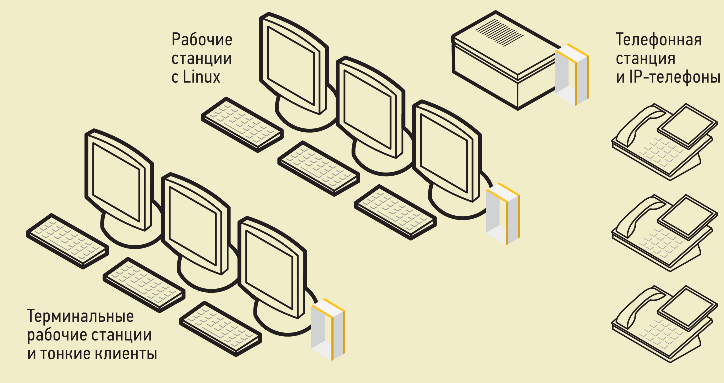
Терминальные рабочие станции и тонкие клиенты

На базе GNU/Linux и виртуализации*

Внедрение и техническая поддержка силами специалистов ГНУ/Линуксцентра

GNU/LINUX
Терминальный сервер
Клиент 1С:Предприятие под Wine@Etersoft
IP-телефония Asterisk
Почтовый сервер Zimbra
Хранилище документов Alfresko
Терминальный сервер
PostgreSQL
Сервер приложений 1С:Предприятие
Файловый сервер
MDS Контроллер домена
DHCP, OpenLDAP, WebConfig

Почтовый сервер, файловый сервер, контроллер домена, базы данных SQL, 1С:Предприятие, IP-телефония — **и все это на одном физическом сервере в виртуальных машинах!**



ПРЕИМУЩЕСТВА ТЕРМИНАЛЬНОГО РЕШЕНИЯ НА БАЗЕ GNU/LINUX:

- экономия средств на лицензировании ПО
- сокращение затрат на оборудование за счет виртуализации
- единая инфраструктура на базе свободного ПО
- защита от вирусов и других угроз безопасности

Позвоните сейчас!
Назначьте встречу с нашим специалистом!

Офис в Санкт-Петербурге
Лиговский пр., 50, корпус 15
Тел.: (812) 309-06-86,
факс: (812) 640-49-90

Офис в Москве
Красноказарменная ул., 17
(в помещении АТС МЭИ)
Тел.: (499) 271-49-54



* Сравнительная стоимость программного обеспечения в комплексном решении для малого и среднего бизнеса.

Свободное ПО + виртуализация = экономия в 10 раз!



Каждый месяц мы сравниваем тысячи программ — а вы можете отдыхать!

Дистрибутивы-брандмауэры



Мартин Мередит изучает дистрибутивы, призванные защищать вашу сеть.



Про наш тест…

Дистрибутивы для брандмауэров исключительно разнообразны, и нам пришлось быть весьма придирчивыми в выборе.

Мы разработали теоретический сценарий, в котором один и тот же брандмауэр используется и дома, и в офисе, с наилучшей гибкостью для обоих мест, и особо следили за легкостью установки и настройки.

Все дистрибутивы, представленные в нашем Сравнении, тестировались на двух системах. Первая – сервер Dell PowerEdge R210, а вторая – виртуальная машина, выполняющаяся в *VirtualBox* на машине HP.

У обеих тестовых машин было по 40 ГБ свободного пространства на жестком диске, и 2 ГБ оперативной памяти.

Наша подборка

Вы идете по темному переулку поздно ночью, и вдруг кто-то наскакивает на вас и отнимает паспорт, кредитные карточки и ключи от автомобиля. Это хорошая аналогия с использованием Интернета. Опасность таится в каждом углу, и при современных соединениях типа «всегда-на-связи» вы можете стать жертвой интернет-эквивалента грабителей, сами того не заметив.

Брандмауэр играет для безопасности вашего компьютера ту же роль, что и дюжий телохранитель, сопровождающий вас на улице. Будь то домашняя или офисная сеть, брандмауэр (он же межсетевой экран) призван обеспечить необходимую защиту.

Большинство брандмауэров делятся на два типа. Первые работают на локальной машине, охраняя вас от ужасов Интернета, на какие бы сайты вас ни занесло. Дистрибутивы Linux, как правило, имеют в своем составе *Iptables*, индивидуально настроенные под данный компьютер или сервер (более подробную информацию о том, как обезопасить свою машину, см. статью *Безопасность: Для вашего сервера* в **LXF121**). Брандмауэры второго типа помещаются между вашей сетью и Интернетом (или другой сетью). Такие-то мы и рассмотрим в нашем Сравнении.

Учитывая огромное разнообразие существующих на сегодняшний день решений (полный список см. на http://linkpot.net/exercised), мы попробуем разобраться, почему одни варианты лучше других, и что больше подойдет вам и вашей системе.

> «Брандмауэр для безопасности компьютера играет ту же роль, что и дюжий телохранитель.»

ClearOS

Дистрибутив, совместивший удобство и функциональность.

ClearOS, безусловно, самый стильный брандмауэр этого Сравнения. На разработку интерфейса явно потрачено много времени и сил, причем большая их часть ушла на вопросы удобства использования.

Поскольку большинство дистрибутивов-брандмауэров написаны для стереотипных технарей, приятно видеть ободряющие перемены в том, что, кажется, уже стало стандартом *де-факто*: «сначала сляпать, потом вспомнить про интерфейс».

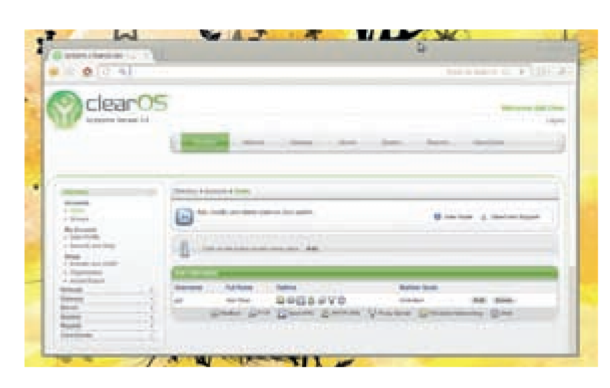
Сайт ClearOS утверждает, что установка занимает всего 2 минуты. Наши тесты обнаружили, что это время ближе к 15 минутам, но сам процесс был безболезненным и простым, и опять-таки, удобство использования инсталлятором хорошо продумано.

Когда вы закончите с этим и перейдете в систему web-администрирования, знакомство с системой не отнимет у вас много времени, благодаря ее простоте. Настройка правил брандмауэра осуществляется быстро и безболезненно, как и значительная часть других действий.

Наиболее заметной особенностью ClearOS является его удобство, но этот дистрибутив дает гораздо больше, чем просто красивый интерфейс. Он предлагает массу функций и не только дает вам простой и понятный путь к управлению брандмауэром, но и позволяет добавить к вашей сети дополнительные сервисы. Это означает, что если вас утомил Windows-сервер, сидящий в углу и рулящий всеми файлообменными службами и заданиями для принтера, вы можете заменить его системой ClearOS.

В целом, ClearOS – мощный дистрибутив, основанный на корпоративных стандартах и обеспечивающий инструменты, необходимые для нормальной работы вашей сети, а также возможность нарастить функционал при возникновении необходимости.

> «На интерфейс ClearOS явно потрачено много времени и сил.»



› ClearOS прост в настройке, а его функционал может быть подогнан под задачи конкретного офиса.



ClearOS Enterprise 5.1

Цена: Бесплатно
Сайт: www.clearfoundation.com/Software/overview.html

» Хорошо продуманный дистрибутив, освежающе простой в использовании и расширяемый под ваши конкретные задачи.

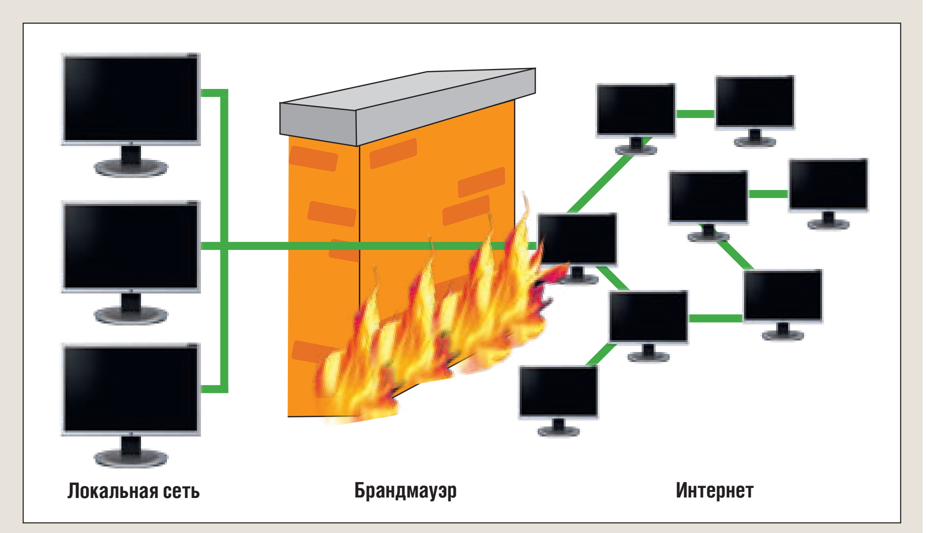
Рейтинг 9/10

Что должен уметь дистрибутив-брандмауэр?

Межсетевые экраны, как правило, располагаются между вашей локальной сетью и Интернетом. Большинство из них также реализуют необходимую сетевую инфраструктуру. Сюда входят DHCP, который выделяет каждому из компьютеров в вашей сети IP-адрес и рассказывает им, как подключиться к Интернету, и DNS, превращающий имена вроде www.linuxformat.ru в IP-адреса (**88.212.205.226**). Дистрибутивы-брандмауэры также дают возможность блокировать входящие соединения на ваши машины и ограничивать интернет-сервисы, к которым предоставляется доступ локальным компьютерам.

Хотя в большинстве дистрибутивов брандмауэры предлагают только базовый набор функций, некоторые заходят дальше и обеспечивают дополнения, укрепляющие безопасность сети либо добавляющие удобства. Некоторые даже позволяют разделять сети на части с различными правилами доступа: например, безопасности ради вы можете вынести бухгалтерию вашей фирмы в отдельную сеть, но разрешить ее сотрудникам печатать на принтеры основной сети. А некоторые сверх того предоставляют различные сервисы, типа баз данных и web- и почтовых серверов.

Еще одна функция, которая нам очень нравится – это одновременная поддержка нескольких внешних сетей (WAN). То есть можно обрабатывать более одного соединения к Интернету (скажем, несколько линий ADSL), распределяя нагрузку между ними или перенаправляя трафик через различные исходящие соединения. Это важно для офисов с несколькими подключениями к Интернету, чтобы не создавалось узкого места.



› Брандмауэр сидит между вашей локальной сетью и Интернетом; его можно настроить на разрешение или блокирование входящих и исходящих соединений.

IPCop

Разносторонний и молниеносный цветастый дистрибутив.

Этот дистрибутив многие превозносили как «Убийцу Smoothwall». Работая по аналогии с Smoothwall Express, IPCop использует для обозначения различных соединений цвета: зеленый для локальной сети, красный для Интернета, оранжевый для DMZ, синий для беспроводных клиентов. По сути, IPCop – ответвление Smoothwall, и вы, вероятно, найдете в них много общего.

IPCop отделился от Smoothwall в 2002 году, и с тех пор набрался сил. Установка очень простая и легкая; новичка озадачат разве что несколько неожиданных вопросов. Однако принятие параметров по умолчанию не вызовет проблем (если у вас сеть без экзотических закидонов).

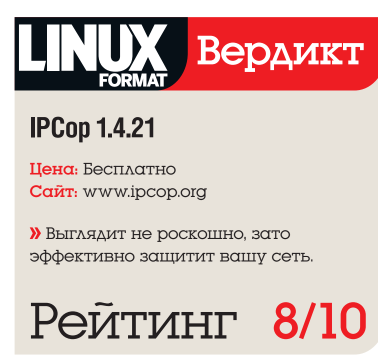
Web-интерфейс IPCop кажется неуклюжим, но наши тесты показали, что это был лишь психологический момент, потому что на самом деле он невероятно отзывчив. И при этом, кроме графиков в режиме реального времени, как в Smoothwall, IPCop дает обильную информацию о параметрах вашей локальной сети, а также о поведении самого брандмауэра, в том числе список соединений, открытых в данный момент.

IPCop обладает функциональностью, полезной, если вы все еще используете модемное подключение к Интернету: здесь можно завести отдельное имя пользователя и пароль, но без доступа к изменению прочих параметров брандмауэра. Он также предоставляет «кэширующий прокси-сервер», так что вы можете сохранять часто посещаемые страницы локально.

Как брандмауэр, IPCop делает хорошую работу, выдавая кучу информации о трафике в сети; быть может, он и не первый красавец среди дистрибутивов, но отвечает своему назначению.

> «Интерфейс кажется неуклюжим, но на деле невероятно отзывчив.»

› IPCop не самый миловидный дистрибутив нашего Сравнения, но службу знает.



IPCop 1.4.21

Цена: Бесплатно
Сайт: www.ipcop.org

» Выглядит не роскошно, зато эффективно защитит вашу сеть.

Рейтинг 8/10

---

eBox Platform

Больше, чем просто брандмауэр.

EBox Platform позиционируется своими создателями не как брандмауэр, но как «Linux-сервер для малого бизнеса» – и он, безусловно, тянет на таковой. Он построен на Ubuntu Server 8.04, так что установка eBox в вашей системе очень схожа с обычной установкой Ubuntu.

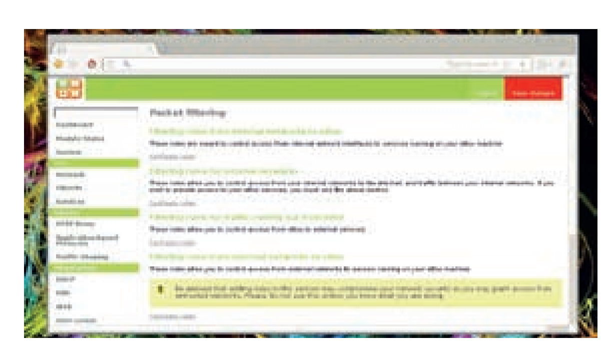
Можно также установить различные компоненты eBox на версию Ubuntu LTS, просто добавив репозиторий *APT* и скачав нужные пакеты – это полезно, если у вас уже есть машина с Ubuntu или если вам требуются лишь определенные части eBox Platform (*ebox-network* и *ebox-firewall*, например). Дело в том, что eBox построен на фундаменте Ubuntu Server и внутри использует его компоненты. Для более подробной информации о различных способах установки eBox Platform, загляните на http://linkpot.net/elixirs.

После установки вы заходите в eBox через ваш браузер, используя пароль, который вы себе задали. На этом этапе вас может ужаснуть чудовищное количество возможностей, предлагаемых eBox. Однако пугаться не следует: стоит лишь отыскать экран брандмауэра, и настройка будет простой.

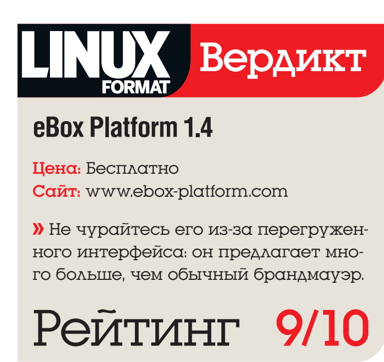
eBox Platform – один из самых увесистых брандмауэров нашего Сравнения с точки зрения размера для скачивания, но не забывайте, что он упаковал множество функций, в том числе для работы с базами данных и серверами SIP, хотя аутентификации Active Directory и PPTP тут нет.

Мы также добавили баллов eBox Platform за широкий охват настроек и за то, что он предоставляет практически все, что требуется для обеспечения работы домашней или офисной сети, прямо «из коробки».

> «Обеспечивает практически все, что требуется для запуска в сети.»



› Как только вы доберетесь до экрана настройки, все понять будет просто.



eBox Platform 1.4

Цена: Бесплатно
Сайт: www.ebox-platform.com

» Не чурайтесь его из-за перегруженного интерфейса: он предлагает много больше, чем обычный брандмауэр.

Рейтинг 9/10

M0n0wall

Встречайте самый крошечный дистрибутив в нашем Сравнении.

M0n0wall – BSD-брандмауэр, предназначенный для работы на 16-МБ флэш-карте, и самый крошечный межсетевой экран из всех, что мы тестировали. Поэтому m0n0wall обеспечивает только самые основные функции. Но при подобной миниатюрности уже и это впечатляет.

После загрузки m0n0wall вы увидите меню настройки. Сперва нужно определить сетевые интерфейсы с функцией m0n0wall 'Auto Detect', которая выручит тех, кто не в силах разобраться, какой идентификатор соответствует каждому сетевому подключению: она позволяет назначать LAN/WAN интерфейсы, выявляя, когда сетевой кабель отключен, а затем подключен снова.

Преимущество m0n0wall в том, что это один из немногих опробованных здесь брандмауэров, который по умолчанию обеспечивает маршрутизацию по качеству обслуживания (QoS); благодаря чему можно следить за трафиком вашего соединения и наделять определенные запросы приоритетами. Это пригодится, если вы хотите использовать VoIP для телефонной связи: удобно будет пустить VoIP с приоритетом.

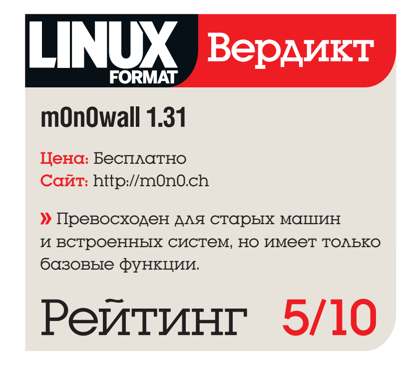
Назначив сетевые интерфейсы, вы можете установить пароль для доступа к системе WebGUI, который позволит вам задать все остальные настройки брандмауэра через web-интерфейс.

Брандмауэр основан на BSD, поэтому некоторые термины могут вначале несколько запутать, но после web-поиска, а также некоторой привычки, он становится понятнее.

M0n0wall мал, да удал: безопасностью он не поступается. Это особенно хорошо для тех, кто хочет получить надежную сеть, не тратя слишком много денег на оборудование, так как он будет прекрасно работать на стандартном, типовом ПК.

> «M0n0wall мал, да удал: безопасностью он не поступается.»

› Пусть m0n0wall и невелик — внутри него много чего настраивается.


LINUX FORMAT Вердикт

m0n0wall 1.31

Цена: Бесплатно
Сайт: http://m0n0.ch

» Превосходен для старых машин и встроенных систем, но имеет только базовые функции.

Рейтинг 5/10

PfSense

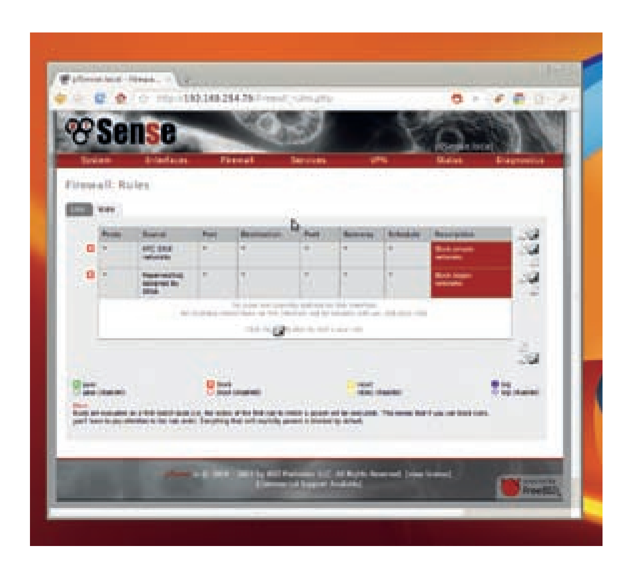
Кому нужен только полноценный брандмауэр и ничего кроме, дальше не ищите.

Название PfSense на первый взгляд кажется странным, но, когда вы осознаете, что это ответвление m0n0wall, а значит, его основа – BSD, смысл проясняется. BSD использует программу *pf* (пакетный фильтр), примерно такую же, как *Iptables*, хотя некоторые говорят, что она более мощная. Это потому, что *pf* и *Iptables* работают по-разному. *Pf* лучше управляется с правилами, хранящими состояние (извлекая или используя информацию о предыдущих пакетах в потоке), а *Iptables* – с правилами без запоминания (не требующими информации о предыдущих пакетах). В этом смысле, *pf* – чуть более надежная защита, чем брандмауэр, использующий *Iptables*, потому что отслеживание последовательности номеров TCP уменьшает шансы на обман.

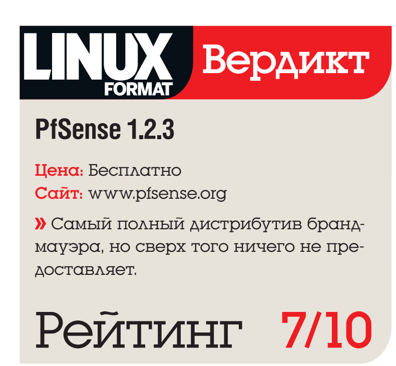
PfSense, как и m0n0wall, прост в установке, которая осуществляется с помощью командной строки; но, в отличие от m0n0wall, интерфейсы требуется создать во время инсталляции, а не после загрузки. И, опять же, определить, какая сетевая карта относится к какому интерфейсу, легко с помощью функции автообнаружения.

Коль скоро это ответвление m0n0wall, вы вправе ожидать, что его возможности будут аналогичны или даже идентичны, но PfSense имеет добавочные функции, такие как мульти-WAN, отказоустойчивость и различные методы аутентификации. Его интерфейс более понятен и удобен в использовании. Опять-таки, поскольку он основан на BSD, некоторые термины сбивают с толку, но вы скоро с ними разберетесь.

PfSense, возможно, богаче всех соперников по функциям брандмауэра, но он проигрывает из-за отсутствия дополнительных возможностей сверх этой конкретной задачи. Если вам просто нужен брандмауэр, вы не ошибетесь, выбрав PfSense, но если вам требуется нечто большее, придется обзавестись еще одной коробкой.



› PfSense — отличный брандмауэр, но ничего, кроме брандмауэра.


LINUX FORMAT Вердикт

PfSense 1.2.3

Цена: Бесплатно
Сайт: www.pfsense.org

» Самый полный дистрибутив брандмауэра, но сверх того ничего не предоставляет.

Рейтинг 7/10

Smoothwall Express

Вероятно, самый известный брандмауэр.

Smoothwall, пожалуй, наиболее популярный дистрибутив брандмауэра. Мы проверили это экспресс-опросом 20 гуру Linux, попросив их назвать дистрибутив брандмауэра. Девятнадцать из них первым назвали Smoothwall.

Инсталляция Smoothwall Express также очень проста, хоть и слегка озадачивает. Определенно стоит скачать пошаговое Руководство по установке [Installation Guide]. В большинстве случаев можно положиться на параметры по умолчанию, и в сети с обычной конфигурацией все должно просто работать.

После первоначальной настройки Smoothwall Express вы можете без проблем запускать его, поскольку он не требует дополнительных действий, кроме подключения сетевых кабелей. Панель управления, основанная на web, проста и легка для понимания. Она дает вам быстрый доступ к функциональности, обеспечиваемой Smoothwall.

Продвинутых функций у Smoothwall Express немного, как видно из нашей таблицы на смежной странице. Однако, подобно IPCop, наряду с кэширующим web-прокси он позволяет иметь отдельную учетную запись, с помощью которой можно контролировать основное соединение, что особенно полезно, если у вас подключение к Интернету через модем. Одно из преимуществ Smoothwall Express – простота при работе с внутренними DNS: добавление нового хоста занимает всего несколько секунд.

Единственная проблема, с которой мы столкнулись во время тестирования – присвоение статического DHCP требует нажатия Добавить [Add], а потом Сохранить [Save], и не особо очевидно, зачем это делать в два шага. У нас это привело к немалой путанице с сетевыми принтерами, которые прыгали с одного IP-адреса на другой.

> «Панель управления на базе web проста и легка в понимании.»



› Smoothwall Express предлагает простой процесс настройки через браузер.



Smoothwall Express 3.0

Цена: Бесплатно
Сайт: www.smoothwall.org

» Прекрасный брандмауэр, прост в использовании, но ему не хватает продвинутых функций.

Рейтинг 8/10

Smoothwall Advanced

Платный брандмауэр – но за ваши денежки он уж попляшет.

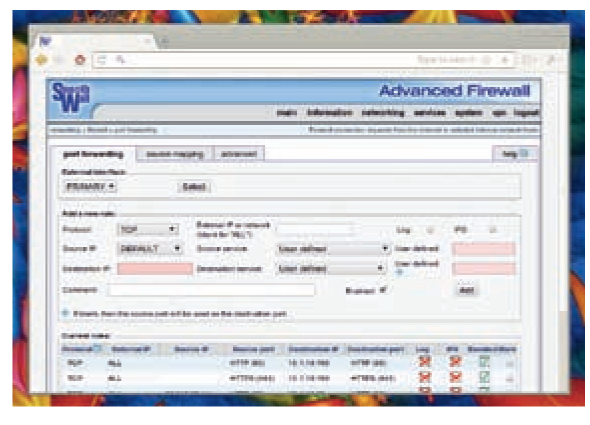
Это единственный платный брандмауэр, попавший в наше Сравнение. Процесс установки Smoothwall Advanced схож с таковым в Express, но вначале потребуется настроить внутреннюю сеть, чтобы получить доступ к его web-панели управления для настройки всего остального.

Smoothwall Advanced кажется созданным для использования в качестве корпоративного брандмауэра, с параметрами для разграничения доступа к различным частям сети на основе аутентификации, web-прокси и системы фильтрации электронной почты. Не то что это плохо, но может оказаться перебором, если вы предназначаете его для дома или малого офиса.

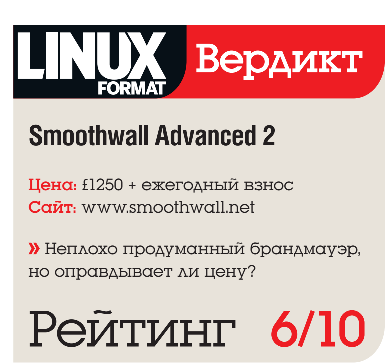
Тем, кто пользовался Express, внешний вид web-панели администрирования будет знаком, хотя мы нашли, что из-за несколько иного подхода к проектированию некоторые простые задачи оказались выполнимы труднее, чем мы ожидали. Но в целом очевидно внимание к деталям, которое характерно и для Express.

Хотя продвинутые функции, доступные в Advanced, идеально подходят для тех, кто хочет решить все проблемы одной коробкой, мы спросили себя, нет ли лучшей альтернативы, и не правильнее ли иметь два компьютера с такой же функциональностью, в целях безопасности. Добавим к этому «ограничения» (допускаются 4 сетевых интерфейса и 20 VPN-соединений), за преодоление которых вам придется приплатить, и Smoothwall Advanced утратит смысл, если только вы не работаете в компании, готовой финансировать ваши нужды, или чувствуете себя спокойнее, пользуясь платным продуктом.

> «Утрачивает смысл вне компании, готовой его оплачивать.»



› Smoothwall Advanced подогнан под корпоративные решения.



LINUX FORMAT Вердикт

Smoothwall Advanced 2

Цена: £1250 + ежегодный взнос
Сайт: www.smoothwall.net

» Неплохо продуманный брандмауэр, но оправдывает ли цену?

Рейтинг 6/10

Дистрибутивы-брандмауэры

Вердикт

eBox Platform 9/10

Выбор правильного дистрибутива брандмауэра во многом зависит от ваших целей. Если вы настраиваете сеть дома или в офисе, установка брандмауэра имеет смысл. По здравом размышлении, брандмауэры являются лучшим способом борьбы с многочисленными опасностями Интернета.

Но в то же время иногда неплохо получить немного дополнительной функциональности, упрощающей жизнь. Затевая наше Сравнение, мы думали, что с победителем все ясно заранее. Но лишь до более внимательного рассмотрения новинок. К своему приятному удивлению, мы нашли некоторых относительных новичков в этой области, но уже с великолепной функциональностью.

Всего лишь брандмауэр

Если вам нужен просто брандмауэр, подойдут все представленные здесь дистрибутивы; одни будут более эффективны, другие менее. В таком случае выбирайте PfSense. Не нравится – тогда IPCop и Smoothwall Express будут отличными вариантами, если вы не любитель сложностей, а Smoothwall Advanced отстает здесь только из-за цены. Если вы хотите сэкономить место или работать на встроенных устройствах, идеальным выбором будет m0n0wall.

Бонусы

Для нас, однако, компьютер, возможности которого не используются в полной мере – деньги на ветер (именно поэтому мы любим виртуализацию). Поэтому нашим победителем стал eBox Platform. Ошеломляющий список функций и тот факт, что он построен на базе стандартной установки Ubuntu, означает, что наряду с брандмауэром у вас будет машина, способная сделать практически все, что можно вообразить.

Правда, нам было довольно трудно сделать выбор между eBox Platform и ClearOS. Но хотя ClearOS предоставляет и множество функций, и удивительно простой интерфейс, eBox имеет больший потенциал по расширению функциональности.

Если вам не нужны все продвинутые настройки, предлагаемые eBox, вы увидите, что ClearOS предоставляет все необходимое в одном, прекрасно сделанном и пригодном для использования пакете.

Наконец, Smoothwall Express заслуживает особого внимания, потому что это единственный брандмауэр, который после установки можно предоставить самому себе, не перебирая настройки для его запуска и работы. А если вам понадобится отыскать в нем конкретные параметры, вы их запросто найдете.

Собственно говоря, до тестирования других дистрибутивов брандмауэров для данного Сравнения, Smoothwall Express, вероятнее всего, стал бы нашим выбором номер один. LXF

› eBox имеет все, что вам может понадобиться, и более того.

> «Было довольно трудно выбрать между eBox Platform и ClearOS.»

Обратная связь

Стоит ли полагаться на брандмауэр для обеспечения нашей безопасности? Или лучше позаботиться о такой защите наших компьютеров, чтобы нам и самим было трудно в них войти? Нашли ли вы дистрибутив брандмауэра для себя, или вы не согласны с нашим рейтингом? Отправьте ваше мнение нам по адресу letters@linuxformat.ru.

Таблица характеристик

Название	Цена	Размер для скачивания, МБ	Мульти-WAN	Web-прокси	Внутренний брандмауэр	Отказоустойчивость	VPN	Аутентификация Active Directory	Антиспам	Почтовый сервер	Контроллер домена Windows
ClearOS	Бесплатно	690	✔	✔	✘	✘	PPTP Open VPN & IPSec	✔	✔	✔	✔
IPCop	Бесплатно	50	✘	✔	✘	✘	IPSec	✘	✘	✘	✘
eBox Platform	Бесплатно	623	✔	✔	✔	✘	OpenVPN	✘	✔	✔	✔
m0n0wall	Бесплатно	18	✘	✘	✔	✘	IPSec	✘	✘	✘	✘
PfSense	Бесплатно	55	✔	✘	✔	✔	IPSec	✔	✘	✘	✘
Smoothwall Advanced	£ 1250 + ежегодный взнос	177	✔	✔	✔	Опционно	SSL L2TP & IPSec	✔	Опционно	✘	✘
Smoothwall Express	Бесплатно	81	✘	✔	✘	✘	IPSec	✘	✘	✘	✘

ПРОБУЙ ЛЮБОЙ!

Грэм Моррисон нацепил виртуальные наушники, чтобы познакомиться с новейшими релизами дистрибутивов без вреда для своей непорочной установки Fedora.



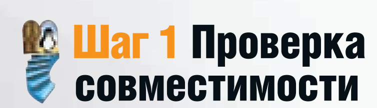
Виртуализации страшиться не надо. Это не монополия предприятий, облачных вычислений или ферм серверов. Она так же полезна и поддается управлению, как и средний настольный компьютер, и сейчас, кажется, способов виртуализовать Linux столько же, сколько дистрибутивов. Например, за деньги можно купить решение для настольной виртуализации либо от *VMware*, либо от *Parallels*, и оба имеют превосходную производительность, поддержку и продвинутые функции. Можно попробовать их открытые эквиваленты — чудесные *VirtualBox* и *Qemu*.

Однако есть и другая возможность, обеспечивающая более прозрачную интеграцию виртуализации в готовую конфигурацию и идеальный способ для экспериментов с новыми дистрибутивами и их тестирования. Это *KVM*, виртуальная машина, встроенная в ядро Linux. Когда-то, быть может, эти три буквы пугали вас из-за слухов о сложности и VNC-сеансах; однако благодаря прекрасному проекту от Red Hat под названием *Virt-Manager*, сейчас почти каждый, при наличии нужного оборудования, может установить *KVM* на свой компьютер и мигом заставить свои виртуальные машины работать.

А виртуальные машины – действительно самый удобный способ экспериментировать с изобилием имеющихся дистрибутивов Linux. Они ничего не портят, их легко установить, а по скорости они почти не отстают от настоящих машин. Это лучший способ прочувствовать дистрибутив, не устанавливая его на реальном оборудовании, и виртуализация позволяет провести столько установок, сколько вы сумеете скачать с Интернета. Как с любым великим путешествием, все начинается с первого шага...

Пошаговая установка

Получите идеальное KVM-решение для виртуализации, следуя нашим инструкциям.



Шаг 1 Проверка совместимости

Прежде чем идти дальше, убедитесь, что ваше оборудование годится для запуска других операционных систем в виртуальной машине *KVM*. В общем-то справиться должен любой компьютер, выпущенный за последние три года, однако следует обратить внимание на некие специфические особенности. Главное, чтобы вашим CPU поддерживались так называемые расширения для виртуализации. Их используют (или могут использовать) все современные гипервизоры (или виртуальные машины), чтобы резко повысить производительность, хотя иные (так называемые «полные виртуализаторы») вроде *VMware Player*, *VirtualBox* и тем более *Qemu*, согласятся работать, даже если этих расширений у вас нет.

Название искомого расширения зависит от типа вашего процессора. Например, владельцы Intel должны искать VT-x, а владельцы AMD – AMD-V. Проверить CPU на совместимость можно, открыв терминал и набрав **cat /proc/cpuinfo**. Выведется все подряд, и если у вас не одно ядро, не один процессор или включена технология Hyper-Threading, вы увидите, что список многократно дублируется. Просто убедитесь, что для любого из ядер в разделе **flags** присутствует либо **vmx**, либо **svm**. Первое – название расширения для виртуализации от Intel, а второе – его аналог от AMD.

> «Нужно отвести память и место на диске каждому экземпляру.»

Поковыряйтесь в BIOS

Если ни того, ни другого не обнаружилось, но вы убеждены, что ваш компьютер это умеет – дело, может быть, в том, что виртуализация отключена в BIOS. Чтобы попасть в BIOS, требуется перезагрузиться и нажать на комбинацию клавиш, указанную в сообщениях системы, до появления окна *Grub*. Обычно это Delete или F2; иногда F10. Место для данной опции тоже зависит от BIOS, однако искать следует либо в разделе Integrated Peripherals, либо в меню Security.

Наконец, позаботьтесь, чтобы вам хватило памяти и места на диске. Виртуальная машина реально-таки жрет ресурсы: нужно отвести память и место на диске каждому экземпляру, не обделив и родную операционную систему. Дистрибутивы Linux обычно работают хорошо при наличии от 512 МБ до 1 ГБ памяти, так что вам понадобится минимум 1 ГБ, а в идеале – 2 ГБ или больше. То же касается дискового пространства. Стандартная установка обычно занимает не менее 5 ГБ (конечно, это зависит от дистрибутива: например, «крошки» типа Puppy Linux гораздо скромнее); однако если вы хотите использовать виртуальную машину для реальной работы, потребуется запастись местом. »

```
root@ubuntu:~# cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 15
model name      : Intel(R) Core(TM)2 CPU         T7600  @ 2.33GHz
stepping        : 6
cpu MHz         : 2328.632
cache size      : 4096 KB
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 10
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic mtrr pge mca cmov pat pse36 clflush d
ts acpi mmx fxsr sse sse2 ss nx vmx lm constant_tsc up arch_perfmon pebs bts tsc_reliable pni s
sse3 cx16 hypervisor lahf_lm
bogomips        : 4657.26
clflush size    : 64
power management:

root@ubuntu:~#
```

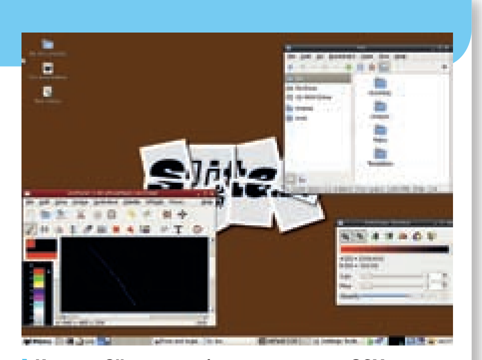
› Чтобы отфильтровать вывод только строк, содержащих «vmx», добавьте «| grep vmx» к «cat /proc/cpuinfo».

А не попробовать ли... Slitaz?

Это фантастический дистрибутив-крошка, и лучшее в нем – его размер. ISO-образ занимает меньше 30 МБ; его можно быстро скачать с ближайшего зеркала, если вдруг потребуется устранение неполадок или быстрый доступ к рабочему столу Linux. Ему удается быть таким маленьким, потому что монолитные программы вроде *OpenOffice.org* заменены в нем на более легкие альтернативы, сохраняющие, тем не менее, достаточную функциональность.

Но это вы увидите после, а сначала придется загрузиться с ISO и, до появления рабочего стола, выбрать раскладку клавиатуры по умолчанию (русская – в нижней половине списка), а также разрешение экрана и глубину цвета. Как и Unity, который мы рассмотрим ниже, Slitaz использует оконный менеджер *OpenBox*, хотя и несколько авангардным способом. Например, по умолчанию на рабочем столе Slitaz находятся только папка My Documents и пара ссылок на документацию и текстовый редактор. И, несмотря на миниатюрность, Slitaz сумел впихнуть на рабочий стол полнофункциональный *Firefox*, с запуском из панели инструментов. Если ваша главная забота – экономия дисковой памяти, есть и другая версия дистрибутива, с еще меньшим размером ISO-образа: *Firefox* в ней заменен альтернативным браузером *Netsurf*.

Замены для общепринятых программ можно найти в меню запуска, и при умеренных запросах для большинства задач этого хватает. Например, *MTPaint* не сравнится с могучим *GIMP*, однако он идеален для быстрых набросков и простого редактирования, примерно как Windows-приложение с похожим именем. Другая прекрасная функция – многоязычный справочник, к которому можно обратиться онлайн через ссылку Documentation. Он содержит обзор различных медиа-форматов, используемых в Slitaz, а также программы и утилиты, входящие в рабочий стол по умолчанию.



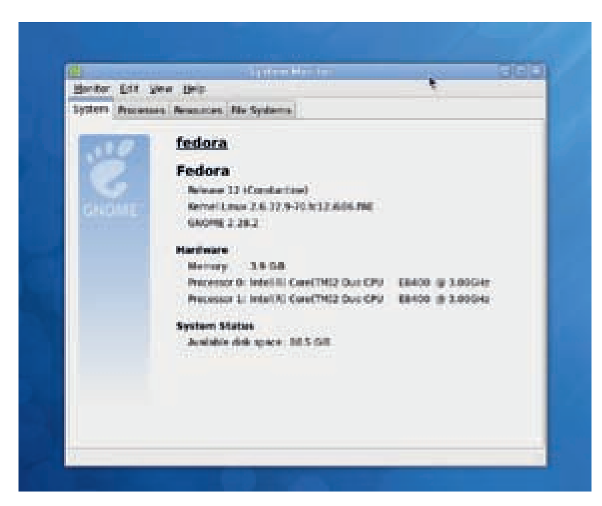
› Крошка Slitaz способен уместиться в ОЗУ, не требуя места на диске.

Шаг 2 Выбор дистрибутива-основы

Мы предпочли Fedora 12: тут удачнее всего реализован *Virt-Manager*, и Fedora 13, являясь адептом, несомненно, будет не хуже. Установить Fedora легко, и это одна из лучших разновидностей Linux.

Virt-Manager – это программа, используемая для управления виртуальными машинами *Xen* и *KVM*, и без нее следовать дальнейшим инструкциям будет намного сложнее. Ведь *Virt-Manager* превращает очень сложную процедуру установки в несколько щелчков мыши. Если вы уже создавали виртуальные машины с помощью коммерческих инструментов вроде *VMware* или *Parallels Workstation*, то увидите, что работа с *Virt-Manager* столь же проста.

Преимуществу Fedora при работе с *Virt-Manager* удивляться нечего – оба созданы в Red Hat. Удивительно другое: нет других дистрибутивов, которые включили бы этот открытый проект в свою стратегию виртуализации. А ведь последние версии этой программы на редкость хороши. Но даже Ubuntu, дистрибутив, сделавший ставку на *KVM* в своей кампании продвижения облачных вычислений с *Eucalyptus*, поместил в свои репозитории пакетов лишь старую версию *Virt-Manager*, сильно уступающую с точки зрения функциональности и удобства. К счастью, в грядущем 10.04 эту проблему рассмотрели, и пользователи Ubuntu скоро смогут установить последнюю версию *Virt-Manager* без больших трудностей.



› **Установка Fedora с 4 ГБ памяти – вот идеальная базовая система для приключений с виртуализацией.**

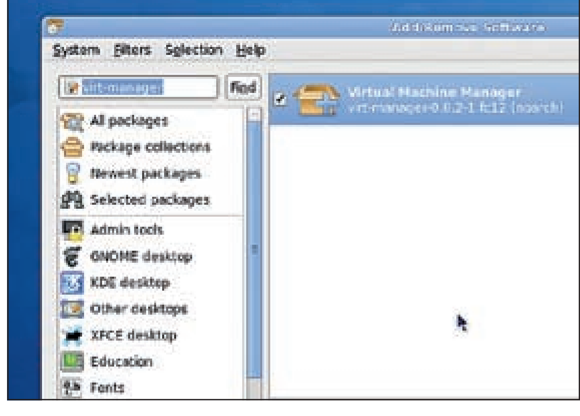
Шаг 3 Установка программ виртуализации

Возможно, самый сложный шаг во всем процессе – установка определенных пакетов, требуемых для виртуализации. Ведь лучшие результаты получаются при использовании командной строки, на которую у многих почему-то аллергия. Но пусть это вас не пугает: мы просто введем строку-другую, и даже с непривычки это не должно вызвать проблем. Fedora лучше всего работает именно таким способом.

Чтобы виртуализация была готова к работе, в Fedora 12 потребуется изрядное количество пакетов. Вы можете использовать менеджер пакетов, запустив его из меню System > Administration [Система > Администрирование] и выбрав пункт Add/Remove Software [Установка и удаление ПО]. Однако у нас возникли сложности при поиске пакета *KVM*; с использованием *Yum* в командной строке получилось лучше.

Из командной строки, которая вызывается из меню Applications > System Tools [Приложения > Системные утилиты], введите **su**, а затем ваш пользовательский пароль. Для установки пакетов наберите **yum install**, а затем имя пакета, который нужно установить. Мы использовали следующее:

```
yum install kvm virt-manager libvirt
```



› **Мы установили *KVM* из командной строки, но утилита Add/Remove Software может показаться вам более удобной.**

Вы увидите, что для удовлетворения зависимостей нужно установить еще и немало других пакетов, однако вам их натаскают автоматически. После установки вы можете либо перезагрузить свою систему, либо ввести **/etc/init.d/libvirtd start** (или использовать команду **service**), чтобы начать процесс управления виртуализацией. И – готово: можно нырять в программу *Virt-Manager*.



Шаг 4 Запуск Virt-Manager

Virt-Manager можно найти, перейдя в System Tools > Virtual Machine Manager [Системные утилиты > Менеджер виртуальных машин]. Чтобы пользоваться программой, потребуется ввести пароль суперпользователя-root, однако при первом запуске много вы не увидите. В главном окне должно быть представлено только одно соединение под названием 'localhost (Qemu)'. В терминологии *Virt-Manager* соединения позволяют управлять виртуализацией, и они могут располагаться как на локальной машине, так и на удаленной. Localhost – это ваш локальный компьютер, а *Qemu* – технология виртуализации, которую использует данное соединение. Выводится *Qemu*, а не *KVM* по той причине, что для обеспечения доступа к стандартным аспектам эмуляции, например, BIOS и различным устройствам, *KVM* использует *Qemu*. Если его не видно в списке, или если вы хотите создать новое соединение, выберите в меню File [Файл] пункт Add Connection [Добавить соединение], а затем в выпадающем списке Hypervisor [Гипервизор] найдите Qemu/KVM. Также можете попробовать *Xen*, если он установлен и вас тянет на эксперименты с другими технологиями (см. **LXF126/127**).

В старых версиях *Virt-Manager* здесь вам пришлось бы вручную создать совместно используемое устройство хранения, щелкнув правой кнопкой мыши по соединению и выбрав накопитель. Новые версии делают это автоматически с помощью мастера подготовки экземпляра ВМ, но это уже следующий шаг.

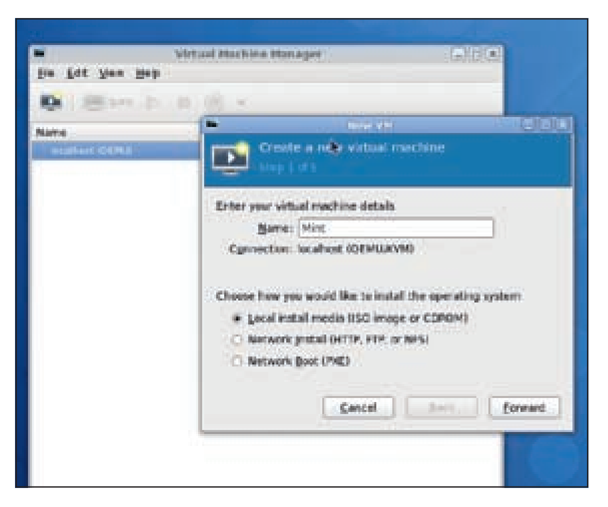
Шаг 5 Создание новой виртуальной машины

Нажмите на значок Play [Запустить] слева вверху и задайте вашему творению имя. Если у вас запущено несколько дистрибутивов, имя виртуальной машины можно дать по названию дистрибутива, который будет в ней выполняться. Не забудьте также выбрать Local Install Media [Локальный установочный носитель]: именно с него вы будете устанавливать дистрибутив.

На следующем экране выберите Use ISO Image [Использовать ISO-образ], и с помощью кнопки Browse [Обзор] перейдите к расположению вашего ISO. Чтобы попасть из виртуального хранилища в свой домашний каталог, нажмите кнопку Browse Local [Локальный обзор]. Виртуальное хранилище – это то место, которое *Virt-Manager* использует для своих виртуальных дисков.

В выпадающем списке OS Type [Тип ОС], расположенном под путем к ISO-образу, выберите Linux. Для наибольшей совместимости, попробуйте указать дистрибутив, наиболее близкий к тому, что вы хотите опробовать. Например, для дистрибутивов типа Mint или Crunchbang лучше выбрать Ubuntu 9.10.

На следующей странице выбирается, сколько ОЗУ отвести под виртуальную машину. Минималистские дистрибутивы типа Dreamlinux довольствуются 512 МБ, а то и меньше, а современные дистрибутивы с Gnome или KDE употребят свыше 768 МБ. Чем большей памятью располагает виртуальная машина, тем лучше она работает. Наконец, на следующей странице, убедитесь, что выбрано Enable Storage [Включить хранилище], и нажмите Create A Disk Image [Создать дисковый образ]. Если у вас достаточно места на диске, увеличьте его размер. Оставив нетронутыми установки на последней странице, жмите Finish [Готово].



› Так как мы устанавливаем из ISO-образа со своего компьютера, выбираем здесь Local Install Media.

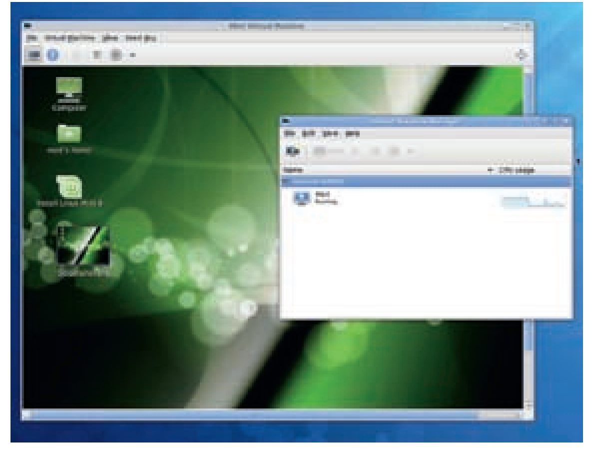
Шаг 6 Загрузка нового дистрибутива



Нажав на Finish, вы заметите, что *Virt-Manager* сразу же начнет процесс загрузки для выбранного дистрибутива. Скоро вы увидите точно те же экраны, что и при старте с настоящего физического диска. Стало быть, все работает как полагается, и вы смогли успешно создать и запустить свою первую виртуальную машину.

Дальнейшее полностью зависит от выбранного вами дистрибутива. Например, Linux Mint выдаст полнофункциональный рабочий стол, а другие дистрибутивы могут попросить вас выполнить процедуру установки. В любом случае, щелкнув внутри окна, вы получите полный доступ к своей виртуальной машине. Затем *KVM* завладеет вашей мышью и клавиатурой. Вы увидите небольшое окно с уведомлением, что курсор захвачен, и показанной комбинацией клавиш, с помощью которой можно выйти из виртуальной машины в исходное окружение.

Обычно такая комбинация клавиш – левые Ctrl и Alt, уже ставшие подобием стандарта для приложений виртуализации. Их одновременное нажатие вернет вас на обычный рабочий стол. При желании передать виртуальной машине определенную клавиатурную комбинацию воспользуйтесь меню Send Key, где перечислены разнообразные сочетания клавиш: например, Ctrl+Alt+F1 для перехода в первую текстовую консоль или Ctrl+Alt+Backspace для перезапуска X-сервера.



› Ура! Заработала! Чтобы покинуть виртуальную машину, нажмите левые Ctrl+Alt.

»

А не попробовать ли... Dreamlinux?

Dreamlinux существует в редакциях с *Xfce* и с Gnome, и последний примерно на 200 МБ больше объема CD-болванки. По этой причине мы взяли *Xfce*, хотя при загрузке дистрибутива в виртуальной машине ограничения на размер ISO нет. Основной интерфейс построен на плавно раскрывающейся панели инструментов внизу экрана, а меню приложений и менеджер задач встроены в его верхнюю часть.

Кнопки запуска теснятся вокруг логотипа Dreamlinux на фоне рабочего стола, и их названия в основном говорят сами за себя. Но есть и расширенные функции, в том числе утилита *MKDistro* для создания собственного дистрибутива Linux и два значка Flash – новинки версии 3.5 (вышла в марте); и создается либо Live CD Dreamlinux, либо Live USB с предустановленным DreamLinux.

Простая установка

Пакеты для многих функций уже сидят внизу экрана. Поскольку DreamLinux основан на Debian, вы увидите менеджер пакетов *Synaptic*, предоставляющий доступ к тысячам пакетов. Как и в Ubuntu и Debian, через него можно добавить любую программу. Но есть и другой способ – это одна из лучших черт DreamLinux. Он называется «Easy Install» и позволяет ставить лучшие программы для рабочего стола Linux, не задумываясь о зависимостях и менеджере пакетов. В постоянной установке просто дважды кликните по значку Easy-Install на рабочем столе. После этого вы одним щелчком мыши сможете установить все что угодно, от *Picasa 3* и *Azureus* до *Songbird*, *Google Earth 5.0* и *Skype*.



› Увидев рабочий стол Dreamlinux, мы первым делом удалили несколько значков.

Извлечь из Virt-Manager все

Виртуальные машины умеют не только запускаться.

Так как на бежевом окне вашей виртуальной машины нет кнопок, ее выключение, перезапуск и отключение от сети выполняется изнутри программы. Эти функции можно найти, либо щелкнув правой кнопкой в окне запущенной машины, либо из выпадающего списка панели инструментов. В зависимости от виртуализованного дистрибутива, использование обеих кнопок – и выключения, и перезапуска – должно быть безопасным.

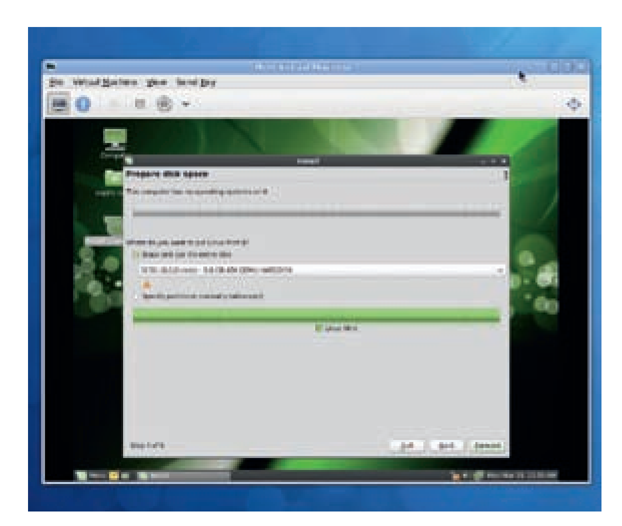
Все потому, что *KVM* посылает сигналы виртуализованной операционной системе, и она должна обработать их точно так же, как обрабатывает выбор команды выключения из меню Gnome или нажатие на кнопку выключения в системе, которая правильно реагирует на сообщения ACPI. То есть вас предупредят о грядущем выключении, давая возможность перейти в работающие приложения и сохранить результат. Такого не случится, если вы выберете в меню выключения пункт Force Off [Выключить принудительно], что является виртуальным эквивалентом выдергивания шнура питания из розетки. Тогда все, что не было сохранено на виртуальном накопителе, будет потеряно.

Вы, наверно, заметили кнопку Pause [Пауза] на панели инструментов *Virt-Manager*. Она мгновенно останавливает виртуальную машину, а повторное нажатие возобновит выполнение с той же точки. Однако в отличие от такой же функции в *VMware*, приостановленная система не выживет после перезагрузки, и данные, которые вы не сохранили в запущенных программах, утратятся.

Осторожно, виртуализация

Важно осознавать, что ваши данные виртуальны, и питание не подводится к каждой из отдельных виртуальных машин; а значит, работа в них намного более уязвима, чем на обычном настольном компьютере. Здесь гораздо больше способов создать неполадку, и восстановление данных тоже более трудоемкое. Если вы обращаетесь со своими данными надлежащим образом, это не проблема, но об этом нужно помнить, если вы принялись засиживаться в виртуальной машине подолгу.

› Вид Details позволяет изменить практически любые настройки виртуальной машины.



› Не бойтесь, создание виртуальных разделов не затронет настоящие на жестком диске.

После первого запуска виртуальной машины вы можете заинтересоваться, как поменять образ диска, чтобы ему соответствовал другой файл ISO, или даже вернуться к той же информации, которая была при первом запуске машины. Эта панель настройки доступна из панели View [Вид] той виртуальной машины, которую вы хотите поменять. Для безопасного применения изменений, эта виртуальная машина не должна быть запущена.

Перейдите в View > Details [Вид > Подробности], и откроется редактор. Вы увидите окно со всесторонним обзором виртуального оборудования, эмулируемого на вашем компьютере. Нажмите, например, на IDE CDROM 1, а затем Connect [Подключить] на панели справа, и вы увидите в виртуальной машине новый образ CD/DVD. Нажмите на Memory [Память], и вы сможете поменять количество памяти, выделенное виртуальной машине. Это очень удобно, если вы недо- или переоценили ее потребности, когда задавали параметры в мастере настройки.

Проверьте свое оборудование

Наверно, вам также небезынтересно, как эмулируется графическое оборудование. Его можно найти на странице Display [Дисплей] – по умолчанию там задано нечто под названием «cirrus». Эмулируемый при этом чипсет Cirrus Logic – один из самых распространенных; его прекрасно поддерживают множество операционных систем. Он идеален при запуске старых дистрибутивов, например, MS DOS и даже Windows, но это не самый быстрый драйвер, и если вы желаете проводить в виртуальной системе достаточно много времени, в списке моделей стоит выбрать «vmvga». Это ближайший аналог графического драйвера, используемого в *VMware*, и для виртуализации он подходит лучше. Если ваш виртуализованный дистрибутив способен использовать реализацию открытого графического драйвера от *VMware*, вы увидите, что эта опция будет работать лучше. Если нет, всегда можно вернуться назад. Последние версии *Virt-Manger* также способны менять масштаб разрешения дисплея виртуальной машины под размер окна. Просто включите опцию Scale To Display > Always [Масштабировать экран > Всегда] в меню View – ее следует за-

А не попробовать ли... Unity?

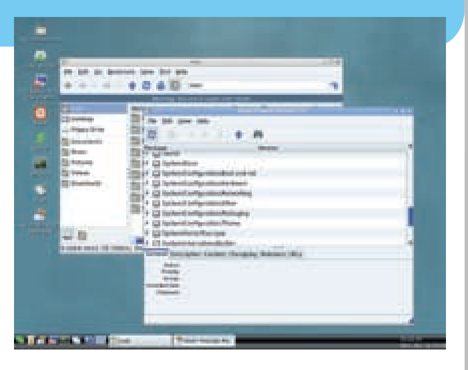
Поскольку виртуализация нужна, чтобы запускать потенциально нестабильные дистрибутивы, можно спокойно экспериментировать с Unity Linux, системой, у которой на момент написания этих строк появился только первый релиз-кандидат. Если с ним возникнут проблемы, или она откажется работать на вашем компьютере, никаких негативных эффектов от этого не будет.

Unity Linux интересен тем, что он основан на Mandriva, дистрибутиве, который сейчас далек от центра внимания. Как многие другие дистрибутивы, Unity Linux поставляется в виде Live CD, крайне удобном для экспериментов. Он довольно сильно отличается от исходного дистрибутива Mandriva и других знакомых нам дистрибутивов Linux. Это становится очевидным даже при взгляде на экран входа, который появится через пару мгновений после загрузки.

Дизайн экрана входа отражает дизайн всего дистрибутива. Он прост и непосредственен. На Live CD, в левом верхнем углу отображаются имя пользователя и пароль для двух активных учетных записей, сиречь «guest» и «guest» для стандартной учетной записи пользователя и «root» и «root» для системного администратора. Если вы намерены установить Unity на жесткий диск, лучше зайти от имени root.

Интересным делает Unity его окружение рабочего стола. Оно построено как набор программ, подобранных вручную по причине функциональности и эффективности и расположенных поверх оконного менеджера *OpenBox*. По сравнению с Gnome и KDE *OpenBox* молниеносен и экономит системные ресурсы. Его системное меню запуска вызывается щелчком правой кнопкой по рабочему столу; есть и минимальное меню запуска, оно откроется при щелчке на листок в левом нижнем углу экрана. Множество утилит – например, меню запуска и файловый менеджер – взяты из *LXDE*, Lightweight X11 Desktop Environment. Они быстрые, и вы вряд ли затоскуете по дополнительным функциям полновесных рабочих столов KDE и Gnome.

Ну, а если все-таки затоскуете – Unity спланирован как стартовая площадка для построения индивидуального дистрибутива. Его менеджер пакетов, вызываемый щелчком на рабочем столе, напоминает старый *Software Manager* от Canonical. Оттуда можно добавить любое количество программ от *LXDE*, Gnome и KDE.



› Пустой на вид, рабочий стол Unity набит возможностями.

действовать, если разрешение экрана виртуальной машины выше, чем у машины-хоста, иначе дисплей придется прокручивать вручную, а это утомительно.

Другая приятная особенность – ваши виртуальные машины доступны и через VNC, протокол удаленного доступа к рабочему столу. Чтобы задействовать его, взгляните на страницу Display VNC [VNC-дисплей] просмотрщика установок. Если виртуальная машина работает, выведется порт, отведенный для службы. После этого вы сможете получить доступ к рабочему столу вашей виртуальной машины, используя клиент VNC – например, *Vinagre* в Gnome или *Krdc* в KDE. Если клиент выполняется на той же машине, просто укажите ему адрес **localhost:5900** для первой виртуальной машины. Если это не сработает, поменяйте номер порта на тот, что был выведен. Вы увидите, что тот же самый сеанс отображается в обзоре виртуальных машин *Virt-Manager*. Использовать это можно в самых разных целях: от доступа к виртуальным машинам с удаленного компьютера, расположенного где-нибудь у Интернета на куличках, до создания двойника рабочего стола для живой демонстрации на проекторе.

> «Виртуальные машины доступны и через VNC-протокол.»

› *Virt-Manager* хорош тем, что для загрузки операционной системы подойдет почти любой ISO-образ.

Продвинутые функции

По дизайну интерфейса *Virt-Manager* вы, наверно, уже догадались, что можно запускать столько виртуальных машин, сколько допускает ваша система. Единственным реальным ограничением будет физическая память – это, вероятно, самое слабое звено системы. При одновременной работе всех машин памяти должно хватать на удовлетворение личных потребностей каждой из них. Например, имея 4 ГБ ОЗУ, можно запустить, наряду с обычным рабочим столом, еще три виртуальные машины, если каждой из них выделить по 1 ГБ, и производительность каждого экземпляра можно проверять, используя CPU-метр, справа от каждого пункта в списке виртуальных машин. Если вам нужно больше информации об использовании памяти, производительности диска и сетевом трафике в виртуальных машинах, зайдите на страницу Performance [Производительность] окна Details [Подробности].

Одна из продвинутых функций *KVM* – предоставление доступа к реальному оборудованию, через функцию Physical Host Device [Физическое устройства хоста]. Но не спешите размечтаться: это не значит, что вы сможете передать виртуальной машине мощную видеокарту Nvidia или звуковое устройство. Слишком уж это сложно. Однако с большинством сетевых адаптеров, а также USB-накопителей, у вас все получится.

Чтобы они заработали, откройте окно Details в панели виртуальных машин и щелкните по кнопке Add Hardware [Добавить оборудование], слева внизу от списка. В появившемся окне выберите из выпадающего списка Physical Host Device, нажмите Forward [Далее] и выберите устройство из выпадающих списков Type и Device. В меню Type задайте тип: PCI или USB, а в Device – выберите подходящее устройство. Эта функция несколько экспериментальная, но вы увидите, что простые устройства будут работать без дальнейших настроек. »

Другие эксперименты

Если ваш CPU не заартачится, найдутся и другие способы.

Виртуализация – не единственный способ опробовать новый дистрибутив; просто так уж вышло, что этот путь самый простой и ненавязчивый. Но при более серьезном тестировании, с запуском нового релиза на реальном оборудовании, есть несколько методов упрощения данного процесса, если вы не хотите портить основную установку.

В отличие от некоторых коммерческих операционных систем (не будем показывать пальцем), большинство дистрибутивов Linux спокойно уживаются с другими дистрибутивами и операционками, автоматически добавляя себя в качестве пункта в загрузочное меню. Было бы место на диске.

Здесь могут начаться фокусы: ведь дистрибутив, установленный первым, будет пытаться захватить все доступное пространство, затрудняя последующие установки. И вы сэкономите свои труды, если ограничите первый дистрибутив указанным количеством места на диске; а для этого потребуется утилита ручной разбивки.

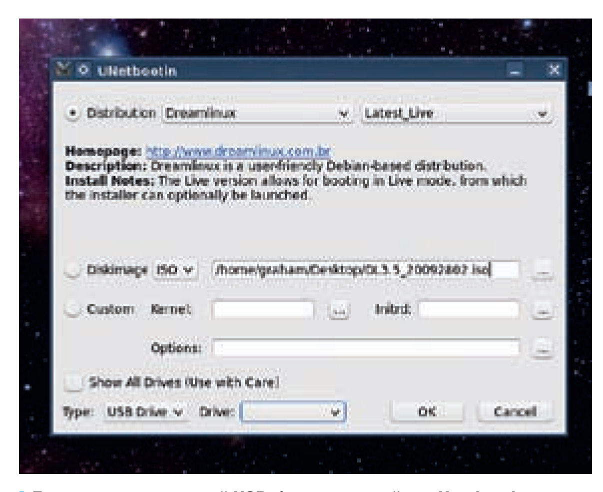
Иди и сравни

Разные дистрибутивы используют разные утилиты для управления процессом разбивки, но базовая функциональность у них одна и та же. Нужно выбирать между автоматической и ручной установками. Первая обычно очищает жесткий диск или же полностью занимает все свободное место, а вторая требует кое-какого ноу-хау.

Как минимум, вам потребуются два раздела: один для файлов дистрибутива, а второй под так называемую подкачку. Если на диске хватает свободного места, вы сможете создать новый раздел и точно указать его размер. Также потребуется выбрать для раздела тип файловой системы; если у вас нет специфических требований, рекомендуем ext4 или ext3.

Раздел подкачки – это область жесткого диска, использующаяся как место для излишних данных в ОЗУ. Эмпирическое правило гласит, что размер этого раздела должен быть вдвое больше размера ОЗУ, но не больше 2 ГБ.

После создания обоих разделов требуется указать каждому точку монтирования. Основному разделу нужно назначить / для корневого каталога, а раздел подкачки обычно отображается как **linux-swap**. Оба следует отформатировать, если у установщика есть такая опция, и тогда процесс установки можно продолжать в обычном режиме. При установке других дистрибутивов процедура будет примерно такой же: разделы для каждой новой установки создаются на оставшемся свободном месте.



Если у вас есть лишний USB-брелок, создайте в Unetbootin дистрибутив, загружаемый с этого устройства автоматически.

Меняем размер из GParted

Если у вас уже установлен дистрибутив, и вы хотите поменять размер текущего раздела, высвободив место для новой установки, эту работу может выполнить *GParted* – утилита, используемая большинством установщиков. Но есть пара оговорок. Наилучшие результаты у нас получались, когда мы загружались с Live CD, содержащего эту программу, например, Ubuntu. Тогда у вас будет полный контроль над вашими дисками, и вы сможете менять размер разделов, не беспокоясь о доступе к данным. Изменение размера при этом сводится просто к выбору раздела, который нужно уменьшить (или увеличить), и нажатию на кнопку Resize. В появ-

А не попробовать ли... Absolute?

Absolute – дистрибутив, основанный на Slackware; данный факт пошлет многих GUI-зависимых пользователей Linux лесом. Однако Absolute не похож на папу, и не нужно быть членом элитного клуба Linux, чтобы он у вас заработал – благодаря включению по умолчанию готового графического окружения, не требующего дополнительной установки пакетов.

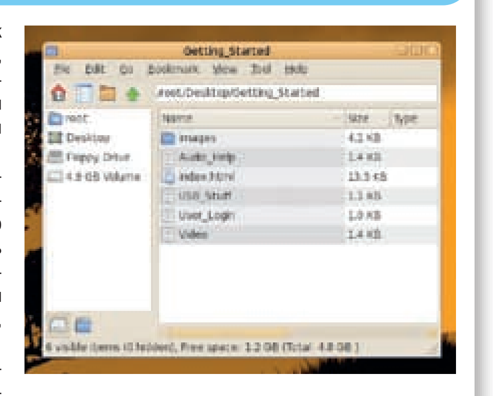
Absolute делает упор на баланс практичности и мощи, присущий Slackware. Родство со Slackware также означает совместимость с широким диапазоном оборудования.

Последняя версия сильно обновлена: добавлены новое ядро, новые компиляторы и последняя версия *GTK*. Ее малозначащий номер (13.1.0) следует схеме нумерации версий Slackware и не отражает важность последнего обновления.

Live-режима нет, и чтобы пользоваться Absolute, вам придется установить его. К счастью, почти все параметры по умолчанию в установщике можно оставить как есть. Например, в первом меню выберите Auto Install, и пусть Absolute сам решает, что делать с жестким диском. Также мы не тронули установки мыши и добавили наши собственные случайные имена хоста и домена для сетевого соединения.

Использование DHCP не вызвало проблем, и мы смогли выйти в Интернет через *KVM*. Споткнулись мы только на том, что на экране входа не отображался курсор и шрифты. Сначала мы подумали, что не распознавалась виртуальная клавиатура, однако если вслепую ввести имя пользователя (по умолчанию подразумевается «root»), а затем пароль, то вы попадете на рабочий стол, и дальше проблем уже не будет.

Рабочий стол прост. Он построен на *IceWM* и укомплектован многими стандартными Linux-приложениями. Он также совместим с пакетами от Slackware, и у вас не будет недостатка программ для установки.



Легкий в установке Absolute поубавил сложностей Slackware.

Редактируем разделы вручную — но сперва сделайте копии!

Изменить/переместить раздел
Выберите раздел и с помощью этой кнопки либо поменяйте его размер, либо переместите на свободное место.

Раскладка
Горизонтальный список показывает физическое размещение разделов в том порядке, как они расположены на жестком диске.

Список разделов
Для каждого раздела отображается файловая система и доступное свободное место.

Выбор устройств
Это выпадающее меню, где можно выбрать либо жесткий диск, либо USB-накопитель.

Свободное место
Если на диске есть свободное место, вы не увидите цветного блока. Двойной щелчок создаст новый раздел.

Изменение размера
Потяните левый или правый край в окне редактирования, чтобы поменять размер раздела.

ляющемся окне перетащите правый или левый край раздела, чтобы уменьшить место, которое он занимает на диске.

После освоения искусства ручной разбивки приходит черед другого аспекта контроля за вашими данными: создание отдельного домашнего раздела. Вам просто нужно создать другой раздел, наряду с корневым и подкачкой, задать ему файловую систему и приписать точку монтирования **/home**.

Большинство установок позволят вам выбрать в качестве домашнего существующий раздел и не потребуют его форматирования. Это значит, что все учетные записи пользователей, а также их данные, останутся нетронутыми и доступными после загрузки нового дистрибутива, что особенно полезно, если вы работаете в нескольких Linux-системах.

Рекомендуем создать для каждого дистрибутива отдельного пользователя. Это устранит накладки в домашних каталогах и файлах настройки, возникающие, если создать пользователя, который уже существует для другого дистрибутива. При желании перенести свои настройки с одной учетной записи на другую, это можно сделать вручную в командной строке, после чего рабочий стол будет готов к работе. Просто скопируйте все содержимое одного домашнего каталога в другой, используя команду **cp -rf источник назначение**, и убедитесь, что права доступа для пользователя и группы соответствуют той учетной записи, которая будет пользоваться этим каталогом (например, с помощью **chmod -R имя_пользователя:имя_пользователя каталог**).

А если вы не любитель ручной разбивки, всегда остается простейшая возможность – старый добрый Live CD. Многие дистрибутивы теперь включают собственные загрузочные версии, позволяя вам протестировать совместимость с оборудованием, а также дизайн и практичность. Вы получите неплохое представление о том, что будет представлять собой конечная установка, хотя чтение данных с оптического носителя ее несколько затормозит.

Дистрибутив на брелочке

Проблема скорости может быть решена, если установить тестируемый дистрибутив на свободный USB-привод и загрузиться с него. Раньше создание таких вот USB-дистрибутивов было тяжким трудом, однако благодаря утилите *Unetbootin* вы сможете создать загрузочный диск почти для всех дистрибутивов всего несколькими щелчками мыши. Пакет *Unetbootin* имеется во многих дистрибутивах, а сама программа, скорее всего, потребует административных привилегий. Когда она запустится, просто укажите расположение ISO-образа вашего дистрибутива и выберите из выпадающего списка его версию, а затем – расположение брелка.

Любой ПК не старше 3–5 лет загрузится с USB-привода без какого-либо вмешательства, однако иногда вам потребуется либо зайти в загрузочное меню из BIOS, либо сменить порядок загрузки в самом BIOS. В результате у вас будет дистрибутив, работающий с внешнего устройства с той же скоростью, что и родной. LXF

Обратная связь

Если вы нашли какой-нибудь чудный дистрибутив, заслуживающий нашего внимания, мы будем рады узнать про это. Напишите нам, чем он хорош, на letters@linuxformat.ru, и мы постараемся уделить ему место в следующем обзоре свободного во всех смыслах мира дистрибутивов Linux.

Ubuntu: Меритократия, а не демократия

По-вашему, нужно лезть со своим мнением, куда захочется? Менеджер сообщества Ubuntu **Джоно Бэкон** объяснит, почему так нельзя…

Наш эксперт

Джоно Бэкон
Музыкант, писатель и разработчик ПО. Он также занимает должность менеджера сообщества Ubuntu, хотя высказанное здесь мнение является его собственным.

Ubuntu имеет много узнаваемых черт, но одна из лучших – репутация его взаимодействия с сообществом. С тех пор, как Марк Шаттлворт [Mark Shuttleworth] создал первую команду в 2004 году, сообщество Ubuntu разрослось в размере, породив разнообразные группы по всему миру. Этот дух сообщества подкреплялся энтузиазмом Марка к приданию прозрачности, внедрению открытого управления и инструментов, кодексу поведения и приглашению добровольцев в ряды проекта.

Недавно, однако, в сообществе возникли определенные разногласия. Они начались, когда Canonical, основной спонсор Ubuntu, объявила об обновлении брэнда по этому проекту. На логотип, обои и многое другое добавились новые краски; новые цветовые схемы, текстуры, обработка фотографий и другие художественные навороты были доведены до сведения широкой общественности. В рамках развития брэнда, ключевые члены сообщества были приглашены в Лондон для работы с группой дизайнеров, а менеджмент сообщества сделал публичное заявление о брэнде.

Объявление, которое я подготовил, включало два экранных снимка, с новыми темами – светлой и темной. Случайный зритель, возможно, этого не заметил бы, но на экранных снимках была деталь, которая кое-кого насторожила: кнопки Закрыть окно / Развернуть / минимизировать переехали справа налево.

Противоречия в сообществе

Перемена была зарегистрирована как ошибка, и чуть ли не каждая собака выступила со своим мнением на этот счет. Некоторые высказывали действительно продуманную критику с точки зрения удобства для пользователя, но многие просто издавали неорганизованные и бессвязные выкрики. Дебаты бушевали до тех пор, пока Марк не вбросил свои пять копеек:

«Мы все делаем Ubuntu, но не все делают его от и до. Иными словами, мы хорошо умеем делегировать. У нас есть команда по ядру, и они принимают ключевые решения о ядре, которые не разрешается принимать людям вне этой команды. Вы можете создавать отчеты об ошибках, комментировать и участвовать, но не оспаривать их решения... У нас предусмотрены процессы, обеспечивающие правильность делегирования, но открытость сообщества отнюдь не означает, что каждый может сказать свое слово обо всем».

Ключевым моментом в ответе Шаттлворта было уточнение, что решения в Ubuntu – не продукт консенсуса: они принимаются уважаемыми и хорошо осведомленными людьми. В заключение он ответил на предыдущий комментарий, подтверждая такую позицию Ubuntu: «Это не демокра-



ся вклад в сообщество Ubuntu

тия. Мы приветствуем обратную связь и информирование, но наши проектные решения принимаются не голосованием».

Через пару секунд после его комментария новостные ленты мира Linux и Open Source бурлили россказнями, что в Ubuntu нет демократии; причем кое-кто, передергивая, утверждал, что способ разработки Ubuntu кардинально изменили. Мой почтовый ящик переполнился.

Когда стряслась эта история, я вспомнил наш разговор с Марком три года назад на саммите разработчиков Ubuntu в Калифорнии. Это был мой первый саммит, и я пока не освоился. Я тогда отвечал за подбор управляемого сообществом Совета для форумов Ubuntu. Мы продумали, что нужно от этого органа, конкретизировали сроки действия полномочий, приняли решения по инфраструктуре и определили, на чем должен фокусироваться совет. Осталось только решить, кто войдет в его состав. В процессе обсуждения подходов к этому делу я предложил провести голосование, на что Марк ответил: «Нет, у нас не демократия».

Сначала моя реакция была той же реакцией кролика-выхваченного-светом-фар, которую недавно выказали некоторые люди. Демократия казалась идейно близким, удобным и справедливым подходом к сообществу, и замечание, что это не для нас, было громом среди ясного неба. Марк уточнил свою позицию: «В Ubuntu решения принимаются не по зрительским симпатиям, а хорошо осведомленными лицами, с богатым опытом по проблеме и принятию решений». Прояснив, чем Ubuntu *не* является, он дал и определение: «Ubuntu – это меритократия».

Рост по заслугам

Для тех, кто не знаком с этим термином: меритократия [от *англ.* merit – заслуга] не предполагает, что право голоса имеет каждый; вместо этого, руководство и управление осуществляют те, кто создал репутацию своими заслугами и хорошей работой. При меритократии по ступеням иерархии поднимаются не благодаря обладанию более шикарной машиной, элегантной одеждой или другими подобными материальными атрибутами. Продвижение идет через большую работу, которая отмечается и вызывает уважение, и основывается на опыте и продуманных суждениях.

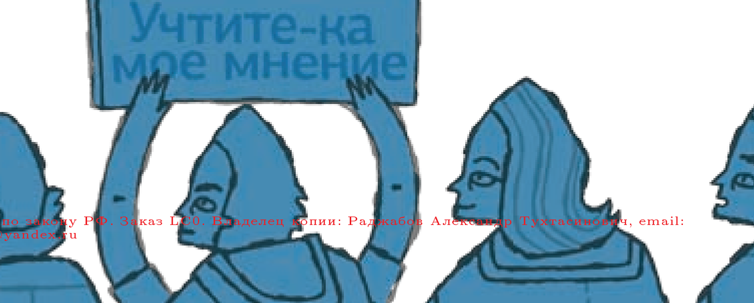
Сообщества, построенные по принципу заслуг, состоят из людей, обменивающихся информацией и сотрудничающих идеализированным образом. В детстве нам прививали, что обход правил и использование статуса или материальных ценностей нельзя применять для достижения успеха в большом мире. С ранних лет нас учили, что добрые дела вознаграждаются добрыми делами.

Сообщества, подобные Ubuntu, работают по тому же принципу. Сообщества, по сути, являются экономиками, но вместо увеличения финансового капитала они развивают свои резервы социального капитала. Мы строим его на основе даров обществу (например, исправлений, документации, сообщений об ошибках и прочего), и когда другие видят наш вклад и уважают нашу работу, мы растем в их сознании как хорошие граждане – граждане, которые имеют опыт и которым мож-

> **«Это не атрибут Ubuntu; большинство сообществ открытого кода меритократичны.»**

но доверить управление. Это не атрибуты, присущие лишь Ubuntu; большинство из сообществ – сторонников открытого кода также меритократичны, и их лидеры определяются по хорошей работе, признанному вкладу и уважению остальными членами сообщества.

Я очень горжусь невероятной работой, проделанной мировым сообществом Ubuntu за последние шесть лет, а меритократия помогла внести в эту работу жизнеспособность, уважение и признание. У нас хватает проблем, над которыми нужно работать и решать их, но наши возможности освещают путь вперед, и я, со своей стороны, готов встать в строй. LXF



Журналы

Раскопайте залежи информации о своей системе



Файлы журналов Linux – что это такое, где их найти и зачем они вам? Ответы знает **Шашанк Шарма**.

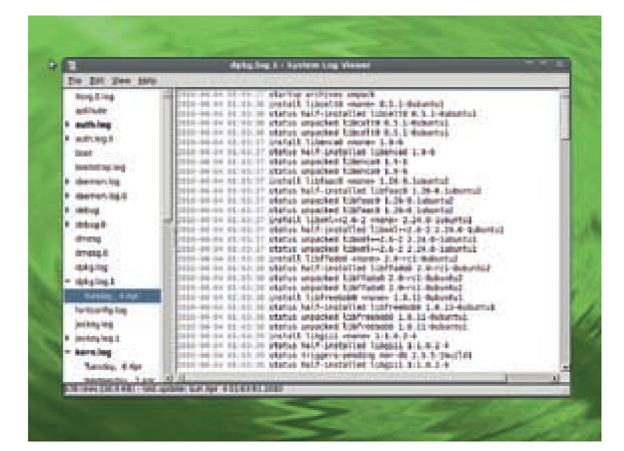
Как заботливый родитель, приглядывающий за занятиями своего чада, Linux бдителен и аккуратно ведет журнал всего, что происходит в системе. Файлы журналов в Linux являются обширным и решающим источником информации при поиске неисправностей вашей системы или отдельных приложений, будь то умолкнувшие беспроводные наушники или проблемная установка *VirtualBox*.

Большая часть журналов помещается в файлах («логах») внутри каталога **/var/log/**. Там находятся самые разные журналы, каждый для своей части системы. Например, есть журналы специально для ядра Linux, равно как для *APT* и *X*-сервера.

Для чтения журналов не требуется никаких специальных программ, ведь они хранятся в виде простого текста. То есть можно не только просматривать лог-файлы, но и создавать скрипты для поиска по ним, а затем выполнять те или иные действия в зависимости от результатов. Это действительно удобно.

> «Почти все журналы в Linux ведутся в одном простом формате.»

Почти во всех дистрибутивах Linux системные журналы создаются соответствующим демоном (*syslogd*). В файле **/etc/syslogd.conf** имеется список всех генерируемых им лог-файлов. Некоторые приложения создают собственные журналы, и они тоже находятся в каталоге **/var/log/**.



› Сжатые журналы не отображаются в стандартном просмотрщике — придется повозиться с прокруткой по горизонтали.

Журналы в Linux

Команда *dmesg* – полезнейшее средство выявления проблем с оборудованием. В основном она выводит на экран информацию обо всех подключенных устройствах, обнаруженных на этапе загрузки. Однако *dmesg* также хранит все сообщения ядра. И если у вас барахлит видеокарта Nvidia, команда **dmesg | grep nvidia** скажет вам, распознала она или нет. Там же ищите сведения про всю свою периферию: web-камеру, USB-перо, накопители, беспроводные клавиатуры и прочее.

Хотя журналы – это простой текст (читаемый и в консоли, и в графических программах), большинство дистрибутивов предусматривает для их просмотра графическое приложение. Пользователям Gnome для этого предлагается утилита *gnome-system-log*. Чтобы увидеть информацию, содержащуюся в журнале, выберите его в списке на левой панели и укажите дату, которая вас интересует.

Из-за громоздкости большинства журналов приходится часто применять программы вроде *grep* и *tail*, чтобы отсеять только нужную информацию, а не прокручивать всякий раз весь файл целиком. Например, использование систем авторизации – программ, которые запрашивают пароль у пользователя, таких как *sudo*, сеансы работы с SSH и тому подобное – отслеживается в файле **/var/log/auth.log**. При активной работе в этом файле может оказаться множество записей – и если вам нужна только информация, связанная с входами через *sshd*, скомандуйте **grep sshd /var/log/auth.log | less**.

По умолчанию, команда *tail* выводит 10 последних строк указанного файла, так что команда **tail /var/log/Xorg.0.log** отобразит последние 10 строк журнала *X.org*. Если их недостаточно для диагностики проблемы с дисплеем, можете использовать опцию **-n**, чтобы *tail* выводил заданное число строк (n) с конца.

Идея применять команду *tail* к файлам журнала состоит в том, что если приложение работает неправильно (например, не появляется GUI), то относящаяся к нему информация будет располагаться в самом конце файла, так как программа уже не выполняется и в журнал не добавляется ничего нового.

Если приложение все же работает и файл журнала обновляется, *tail* все равно пригодится для отслеживания его обновлений в реальном времени, благодаря опции **-f** (follow, следовать). Скажем, команда **tail -f example.log** запустит *tail* в цикле, который будет постоянно ожидать добавлений в файл. Для выхода из цикла и возврата в терминал нажмите Ctrl+C.

Архивы журналов и logrotate

Просмотрев содержимое каталога **/var/log/**, можно заметить, что некоторые лог-файлы имеют имена **kern.log.0**, **kern.log.1.gz** и так далее. Они образуются после ротации (автоматической архивации) журналов. С течением времени почти все журналы изрядно разрастаются, и читать их становится сложно. Тогда первоначальный файл журнала переименовывается, и заводится новый журнал **kern.log**.

Утилита *logrotate*, управляемая файлом **/etc/logrotate.conf**, отвечает за создание новых журналов, а также переименование и архивирование старых. Файл **/etc/logrotate.conf** обычно имеет следующий вид:

```
# rotate log files weekly
weekly
# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create
# uncomment this if you want your log files compressed
#compress

# packages drop log rotation information into this directory
include /etc/logrotate.d
```

Каталог **/etc/logrotate.d** содержит настройки для отдельных файлов журналов: **apt**, **dpkg** и т.д.

При соответствующей настройке *logrotate* не сразу удаляет старые журналы, а какое-то время хранит их в сжатом с помощью *gzip* виде. При желании можно указать, чтобы перед удалением *logrotate* отсылал старые журналы на заданный адрес электронной почты. Но прежде чем браться за реформы в **/etc/logrotate.conf** или любом файле из каталога **/etc/logrotate.d**, изучите man-страницу *logrotate*.

Более важными опциями настройки являются daily, weekly или monthly, которые используются для определения частоты ротации журнала. Опция **rotate count** укажет *logrotate* выполнить ротацию журнала **count** раз перед тем, как удалить его. Так, **rotate 4** будет оставлять на хранении только четыре устаревших журнала.

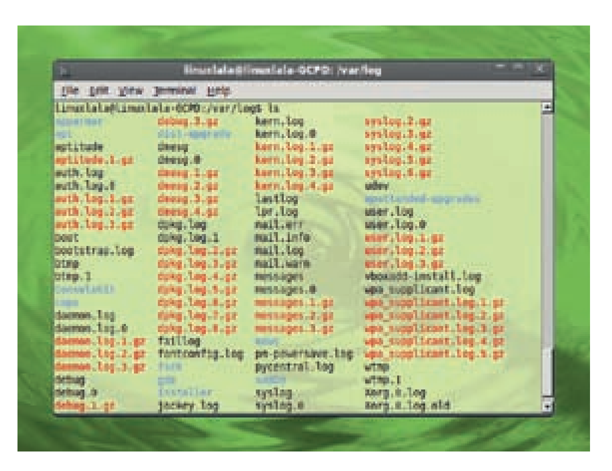
Расшифровка журнала

На первый взгляд строки текста, составляющие лог-файл, кажутся полной бессмыслицей, но на самом деле разобраться в них несложно, и, к счастью, почти все журналы Linux ведутся в одном и том же простом формате.

```
Apr 7 03:50:01 linuxlala-GCPD CRON[5583]: pam_
unix(cron:session): session opened for user root by (uid=0)

Apr 7 03:50:04 linuxlala-GCPD CRON[5583]: pam_
unix(cron:session): session closed for user root
Apr 7 03:55:04 linuxlala-GCPD gdm[2390]: pam_unix(gdm-
autologin: session): session closed for user linuxlala
Apr 8 03:27:08 linuxlala-GCPD gdm[2412]: pam_unix(gdm-
autologin: session): session opened for user linuxlala by (uid=0)

Apr 8 03:27:08 linuxlala-GCPD gdm[2412]: pam_ck_
connector(gdm-autologin:session): nox11 mode, ignoring
PAM_TTY :0
```



› Список журналов в каталоге /var/log/ зависит от дистрибутива. Например, в Fedora есть журналы *Yum*, а в Ubuntu — журналы *APT*.

Каждая строка этого небольшого фрагмента, взятого из файла **/var/log/auth.log**, содержит пять важных кусков информации, в таком порядке:

1. Дата и время, когда сообщение попало в журнал, например **Apr 7 03:55:04**.
2. Имя компьютера (hostname), от которого пришло сообщение – в нашем случае это **linuxlala-GCPD**.
3. Имя программы или службы, например, **gdm** или **cron**.
4. ID процесса программы или службы, пославшей сообщение – в квадратных скобках.
5. Собственно текст сообщения.

Некоторые журналы в каталоге **/var/log** могут читаться только специальными программами или утилитами. Упомянем в этой связи лог-файл отказов входа в систему (**/var/log/faillog**), журнал последних входов (**/var/log/lastlog**) и журнал записей о входе (**/var/log/wtmp**), но их, однако, не прочесть без команд *faillog*, *lastlog* и *who*, соответственно.

Наконец, ниже приведен список наиболее важных и интересных журналов, которые когда-нибудь вам обязательно пригодятся. LXF

Семь полезных журналов

Имя файла	Название журнала	Описание
boot.log	Журнал загрузки	Помимо последних загрузочных сообщений, расположенных в самом конце, содержит информацию обо всех запущенных и остановленных службах.
dmesg	Журнал запуска ядра	Записывает все сообщения, выдаваемые ядром при загрузке (и не только).
messages	Журнал сообщений	Многие программы и службы записывают сообщения в этот журнал общего назначения.
Xorg.0.log	Журнал X-сервера	Сообщения от *X*-сервера *X.org*.
Kern.log	Журнал ядра	Сообщения от ядра Linux. Они помогут исправить неполадки при установке нового ядра.
Syslog	Системный журнал	Содержит информацию почти обо всем. Не найдя нужной информации в других журналах, обращайтесь сюда.
daemon.log	Журнал демонов	Предоставляет сведения о системных и прикладных демонах, таких как служба Bluetooth HCI, база данных *MySQL* и т.д. Полезен при диагностике проблем, связанных с демонами.

Ubuntu Tweak
Мастер настроек

Скучаете по Панели управления? У **Григория Рудницкого** есть для вас достойная альтернатива.

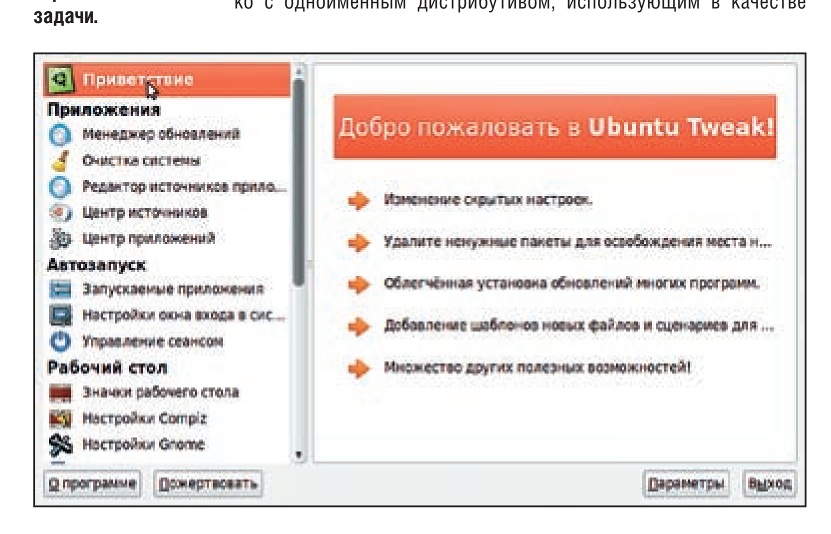
Ubuntu получил известность как система, простая и удобная в установке и настройке, что сделало его открытым и доступным даже для новичков, только начинающих знакомиться с компьютером либо имеющих небольшой опыт работы с Windows или Mac OS X. Действительно, настройки и набор программ, предлагаемые разработчиками по умолчанию, выверены до мелочей. Однако по мере погружения в Ubuntu у всех пользователей рано или поздно возникает желание поменять какие-либо системные параметры, добавить новое ПО и т.д. Конечно, все это можно сделать с помощью служебных программ, сгруппированных в соответствующем меню, или даже путем редактирования конфигурационных файлов. Не знаю, кого как, а меня лично немного нервировало отсутствие в составе Gnome единого центра настройки системы, который испокон веков присутствовал, например, в KDE — я уж не говорю о всеобъемлющих конфигураторах, например, *Yast* из SUSE.

К счастью, сообщество давно решило эту проблему, и тем пользователям, кто ощущает дискомфорт, ползая по меню Параметры и Администрирование в поисках нужного инструмента, можно порекомендовать многофункциональную утилиту *Ubuntu Tweak*. Вы не найдете эту программу в официальном репозитории, но ее всегда можно скачать с сайта разработчика (ubuntu-tweak.com) в виде двоичного пакета, либо просто дописать в файл **/etc/apt/sources.list** строку **deb http://ppa.launchpad.net/tualatrix/ubuntu ваша_версия_дистрибутива main**. Затем понадобится обновить источники, после чего *Ubuntu Tweak* будет доступен через пакетный менеджер.

Важно также отметить, что *Ubuntu Tweak* работает не только с одноименным дистрибутивом, использующим в качестве графической оболочки Gnome, но и с другими, созданными на его основе: Kubuntu, Xubuntu и Lubuntu.

> «Рано или поздно возникает желание поменять параметры.»

› Главное окно Ubuntu Tweak: слева — категории, справа — основные задачи.



Итак, что же умеет Ubuntu Tweak? Очень и очень многое. Интерфейс у этой утилиты многоязычный и автоматически подстраивается под текущие параметры рабочего стола. В основном окне – которое, кстати, отлично выглядит при любом разрешении экрана, в том числе и на нетбуках – представлен краткий перечень задач, которые можно решить с помощью *Ubuntu Tweak*. В левой половине окна функции программы сгруппированы по категориям.

Работа с программами

В категории Приложения находятся все функции, так или иначе связанные с ПО, входящим в состав Ubuntu. В частности, здесь находится удобный менеджер обновлений, который является неплохой заменой стандартному. Здесь все просто и зависит от настроек проверки репозиториев. Если на сервере обнаружены новые версии установленных пакетов, пользователь будет извещен об этом. Кроме того, он может осуществить проверку и обновить систему самостоятельно.

Также в этой категории можно найти пункт, отвечающий за очистку системы. В данном случае, возможности *Ubuntu Tweak* намного шире и корректнее, чем у стандартного пакета *Computer Janitor*. В качестве «мусорных» компонентов последний будет предъявлять пользователю установленные вручную программы, которые либо устарели, либо просто не поддерживаются ни в одном из известных системе репозиториев. В случае с *Ubuntu Tweak* такого не происходит. Процесс очистки здесь распределен по нескольким пунктам. Это удаление пакетов, оставшихся в кэше менеджера *Synaptic* после установки или обновления, удаление «осиротевших» пакетов, оставшихся в системе после деинсталляции программ, очистка от конфигурационных файлов, уже отсутствующих в системе приложений, и, наконец, удаление старых версий ядра, если произошло его обновление. Это будет особенно полезно владельцам нетбуков с небольшим по объему накопителем, для которых освобождение лишних мегабайт является порой просто жизненной необходимостью.

Не менее удобен и Редактор источников приложений, который предоставляет доступ к конфигурационным файлам, содержащим адреса репозиториев, из которых осуществляется обновление и загрузка ПО в системе. С помощью *Ubuntu Tweak* очень легко устанавливать в систему и ПО третьих сторон. Для этого можно использовать Центр источников. Он позволяет подключать в систему новые репозитории буквально одним щелчком; правда, при этом появляется окно с предупреждением о том, что использование стороннего ПО может быть небезопасно. Зарегистрировавшись на сайте Ubuntu-Tweak.com, можно самому добав-

лять новые, в том числе и свои собственные, репозитории. Разумеется, это касается прежде всего разработчиков ПО.

Кроме того, *Ubuntu Tweak* позволяет устанавливать новые программы. За эту операцию отвечает пункт Центр приложений. Конечно, он не заменит традиционный *Synaptic* – список программ в нем намного меньше; зато здесь собраны самые «сливки», то есть наиболее популярные приложения от сторонних разработчиков, для удобства распределенные по категориям.

Управление автозагрузкой

В разделе «Автозагрузка» находятся инструменты, с помощью которых можно управлять процессами, автоматически запускающимися в момент загрузки операционной системы. Каждый процесс снабжен описанием, и вы всегда будете знать, что делаете, и не отключите что-нибудь жизненно необходимое. Помимо отключения ненужных программ, можно добавлять в автоматическую загрузку собственные. Также в данном меню вы сможете настроить окно входа в систему, отключив, к примеру, отображение списка пользователей, если на вашем компьютере работает только один человек, или сменив фон экрана приветствия и даже логотип на нем. Наконец, в данном пункте можно выбрать диспетчер файлов, оконный менеджер и панель, которые вы хотите задействовать в системе по умолчанию.

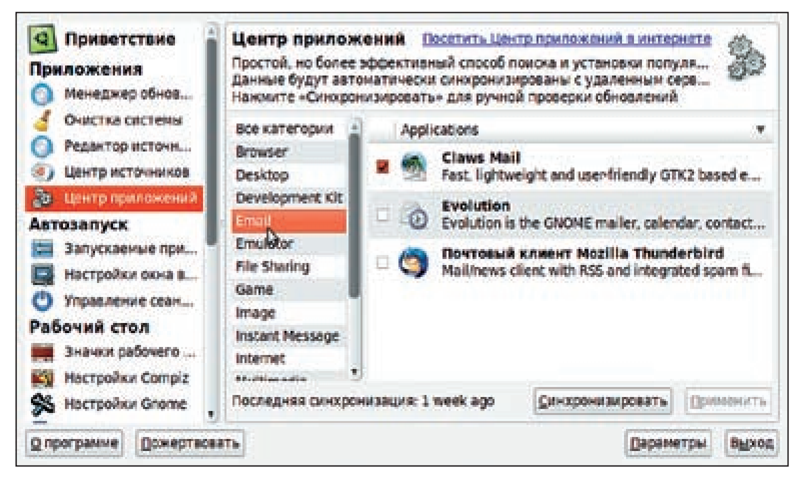
Внешнее убранство

Комфортное, удобное и красивое рабочее место оказывает положительное влияние на производительность труда. В составе *Ubuntu Tweak* имеются инструменты, которые позволяют управлять иконками на рабочем столе, включать или отключать их, а также переименовывать. Если вы любитель красивых эффектов, то вам прямая дорога к менеджеру *Compiz*. Его настройками вы сможете управлять из отдельного модуля *Ubuntu Tweak*, а если *Compiz* и вовсе не установлен, то оттуда же можно подгрузить все необходимое, причем никаких лишних вопросов *Ubuntu Tweak* вам задавать не будет – он просто скачает и установит файлы, а затем отобразит окно настройки. Точно так же можно менять различные параметры поведения графической среды Gnome и оконного менеджера, который вы используете. Единственное, чего здесь не хватает – это кнопки возвращения настроек, установленных по умолчанию. Это было бы полезно для особо увлекающихся личностей.

Полная оптимизация

В разделе Личные сосредоточены инструменты, отвечающие за персонализацию уже не внешнего вида, а внутренней структуры шаблонов, файлов и папок. К примеру, здесь можно изменить пути к основным пользовательским каталогам. Допустим, необходимо, чтобы видео или загруженные из Интернета файлы сохранялись не в домашнем каталоге, где может быть не так уж и много места, а на другом физическом диске, в том числе и съемном. Если последний смонтирован и опознан системой, достаточно просто указать его в качестве места сохранения. Перепутать не страшно, так как в этом случае разработчики предусмотрели возможность отката к значению, установленному по умолчанию. Правда, если вы уже сохранили файлы в новом месте, переносить их в домашний каталог придется вручную.

Еще одна полезная возможность, которую *Ubuntu Tweak* предоставляет для оптимизации вашей работы – это управление сценариями (скриптами). В окне данного модуля вы видите обширный список скриптов, которые можно подключать или отключать простым перетаскиванием мышью. Если тот или иной скрипт подключен, он будет доступен в контекстном меню, и вы сможете им воспользоваться. Наконец, нельзя не отметить работу с шаблонами документов – принцип здесь примерно такой же. Чтобы сохранить тот или иной документ в качестве системного шаблона, его нужно просто перетащить мышью в окно модуля *Ubuntu Tweak*, отвечающего за работу с шаблонами.



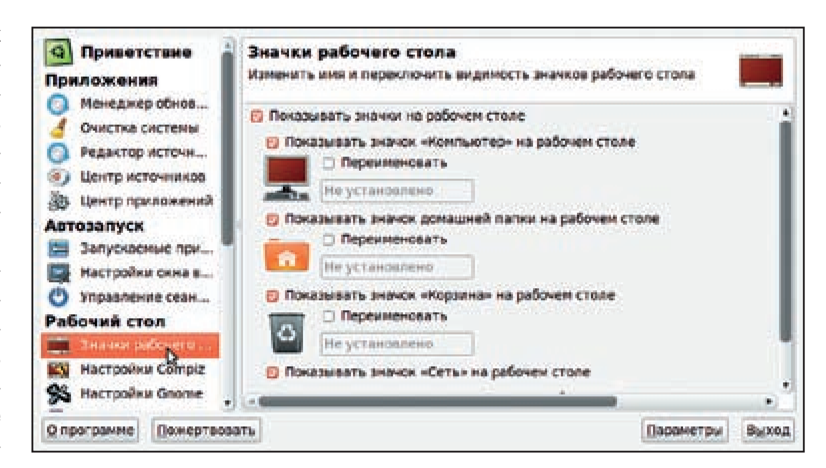
› «Центр приложений» Ubuntu Tweak делает ставку не на количество, а на качество и популярность программ.

Системные настройки

Раздел Система позволяет управлять форматами файлов и привязанными к ним программами. Кроме того, здесь можно менять целый ряд параметров, относящихся к безопасности, таких как запрет сохранения на диск, быстрое переключение между пользователями и т.д. Чтобы поменять эти опции, придется ввести пароль суперпользователя. Отдельно выведены настройки *Nautilus* – изменение размера эскизов, а также включение различных дополнительных возможностей. Мне, к примеру, очень пригодилось расширение для *Nautilus*, отвечающее за открытие любого каталога с правами суперпользователя. И последнее, о чем хотелось бы упомянуть – это управление питанием, что будет особенно полезно владельцам ноутбуков и нетбуков.

А в чем выгода?

«Зачем все это? – могут спросить некоторые читатели. – Зачем повторять открытие Америки и изобретение велосипеда?». Конечно, разработчики *Ubuntu Tweak* не открыли ничего нового, и все вышеописанное можно осуществить и другими способами. Часть из них не уступает по удобству возможностям героя нашего обзора, а для других нужны знания и опыт. Но ведь вряд ли кто-то будет хранить дома инструменты – гаечные ключи, молотки и отвертки – в разных углах квартиры: большинство людей воспользуются специальным ящиком для всего инвентаря. И, как бы то ни было, нельзя не согласиться с тем фактом, что разработчики данной утилиты сделали самый популярный дистрибутив еще ближе и понятнее для многих неофитов в мире Linux. LXF



› Здесь можно указать, какие служебные иконки мы хотим видеть на рабочем столе, а какие — нет.

Живой звук



LINUX FORMAT на DVD
» Сценарий rtirq
» Ядро Linux 2.6.33 с rt-патчем

Желаете использовать Linux в студии звукозаписи? **Дмитрий Михирев** расскажет, как превратить просто свободную ОС в свободную ОС реального времени.



Наш эксперт

Дмитрий Михирев
Достаточно ленив, чтобы потратить уйму времени на адаптацию своей системы под конкретную задачу – лишь бы не пришлось устанавливать специализированный дистрибутив.

Работа со звуком – одна из областей, в которых Linux на высоте. И всё же при необходимости превратить компьютер в импровизированную студию звукозаписи или микшерный пульт непременно возникает ряд проблем: вроде всё необходимое ПО имеется, но настроить его так, чтобы оно работало безупречно, непросто. Это даже послужило причиной появления специализированных дистрибутивов Linux, ориентированных на работу со звуком; наиболее известный из них – 64 Studio (www.64studio.com). И всё же устанавливать отдельную ОС под одну-единственную задачу – это слишком расточительно. Давайте лучше разберёмся, как сделать всё необходимое на базе любого уже установленного дистрибутива.

Зачем это надо?

Чтобы просто слушать музыку в Linux, никаких особых ухищрений не требуется. Необходимость в них возникает при профессиональной работе со звуком, когда компьютеру отводится роль того или иного устройства обработки – будь то микшерный пульт, гитарный процессор или просто записывающее устройство. Основное требование, предъявляемое к этой аппаратуре – отсутствие сколько-нибудь заметных задержек сигнала. Музыкант должен слышать в мониторах ту ноту, которую он берёт сейчас, а не ту, которую извлёк из инструмента полсекунды назад. Такие требования обуславливают неприемлемость сколько-нибудь значительной буферизации, а без неё возможно появление «заиканий» звука, что ещё хуже.

> «Чтобы просто слушать музыку в Linux, ухищрений не требуется.»

Получается, казалось бы, порочный круг, выбраться из которого не так-то просто. Однако у любого современного компьютера вполне достаточно вычислительных мощностей для обработки нескольких аудиопотоков в реальном времени: надо лишь использовать их оптимальным образом.

Как это делается

Чтобы ничто не мешало обработке звука, надо «всего лишь» превратить Linux в систему реального времени. Задача эта непростая, но с ней успешно справился Инго Мольнар [Ingo Molnar], а уже скомпилированные ядра с наложением написанных им патчей есть в репозиториях многих двоичных дистрибутивов.

Потребуется также звуковой сервер, обеспечивающий минимальные задержки, но (при необходимости) позволяющий производить даже многоступенчатую обработку звука. Тут всё очевидно: этим требованиям удовлетворяет JACK, и никаких серьёзных альтернатив ему нет – да и нужны ли они? Поскольку в большинстве современных дистрибутивов по умолчанию устанавливается *PulseAudio*, его нужно будет отключить, чтобы он не оттягивал на себя ресурсы.

Ну и, само собой, потребуется собственно ПО для обработки звука в реальном времени. С этим замечательно справляются как самостоятельные программы, так и многочисленные модули LADSPA и LV2 (LADSPA version 2). JACK позволяет использовать для последовательной обработки звука практически любое их число. В большинстве дистрибутивов, установив пакеты *ll-plugins* и *swh-plugins*, вы получите возможность делать со звуком почти всё, что угодно.

С аппаратной частью обстоит сложнее: требования к ней зависят от конкретных задач. В любом случае, потребуется качественная звуковая карта, а если количество входов на ней мало – придётся раздобыть аппаратный микшерный пульт. Если компьютер должен будет заменить ещё и синтезатор, то, разумеется, не обойтись без MIDI-клавиатуры. Впрочем, вопросов выбора оборудования мы в этой статье касаться не будем, а вплотную займёмся настройкой программ.

Приступим

Ну что же, пора перейти от слов к делу. В первую очередь разберёмся с ядром реального времени. Как я уже отмечал, таковое имеется в репозиториях многих двоичных дистрибутивов. Например, в Mandriva понадобится установить пакет *kernel-rt-latest*. Пользователям Ubuntu следует установить метапакет *linux-rt*, а в других дистрибутивах следует поискать пакет, название которого начинается на *kernel-rt*. Не лишним будет прочитать описание пакета: обычно в ядре реального времени присутствуют некоторые ограничения. Например, в Mandriva оно не содержит никаких дополнительных патчей, и, как следствие,



› *QjackCtl* позволяет запускать сервер JACK с нужными параметрами и управлять им, совершенно не задумываясь о ключах командной строки.

может не поддерживать некоторое оборудование, нормально работающее со стандартным ядром.

Но установка пропатченного ядра – это даже ещё не полдела. Само по себе оно ничего не изменит, поэтому потребуется также скрипт *rtirq*, повышающий приоритет потоков (threads), которые обрабатывают прерывания, назначенные аудиоустройствам. Этот скрипт существует в двух вариантах: в виде обычного исполняемого файла, который можно запустить вручную, и в виде init-сценария, выполняющегося при загрузке системы. Именно второй вариант нам и понадобится. В Ubuntu он уже имеется в репозиториях под именем *rtirq-init*, а пользователи других дистрибутивов могут найти его на прилагающемся к журналу диске.

Ну и, разумеется, надо установить JACK. В дополнение к нему не лишним будет *QjackCtl* – если, конечно, вы не фанат консоли и не предпочитаете запускать JACK с помощью собственноручно написанного скрипта.

Прежде чем перезагрузиться и задействовать, наконец, ядро реального времени, нужно обеспечить себе возможность повышения приоритета процессов: с настройками по умолчанию система может этого не позволить. Первым делом следует убедиться, что пользователь, от имени которого будет запускаться JACK, состоит в группе audio, и если это не так – добавить его туда. Затем членам этой группы надо разрешить повышение приоритета реального времени до 99, а также увеличить лимит блокирования памяти, отредактировав файл **/etc/security/limits.conf** (в некоторых дистрибутивах – **/etc/limits.conf**). Нужно найти строку вроде этой:

```
@audio    -    rtprio    50
```

и исправить её на

```
@audio    -    rtprio    99
```

Кроме того, добавьте строку

```
@audio    -    memlock    1048576
```

Число здесь – ограничение памяти в килобайтах; следует отдать столько, сколько не жалко.

Сделай сам

К сожалению, готовое ядро реального времени имеется не во всех дистрибутивах. Если это про вас, то его придется собирать самостоятельно. Понадобится только исходники ядра (с kernel.org) и патч для той же версии ядра, который можно найти по адресу http://www.kernel.org/pub/linux/kernel/projects/rt/. (Пользователи Gentoo могут просто установить пакет *rt-sources* с уже пропатченным ядром.)

Архив с патчем надлежит поместить в директорию с исходниками ядра и затем наложить командой

```
bzcat ../patch-2.6.23.1-rt11.bz2 | patch -p1
```

Это, конечно, ещё меньше, чем полдела. Далее ядро нужно правильно сконфигурировать. Если в данный момент запущено ядро той же версии, что собираете вы (с точностью до числа после второй точки), можно облегчить себе задачу, используя его конфигурационный файл. Извлечём его –

```
zcat /proc/config.gz > .config
```

и применим команду

```
make oldconfig
```

На все выводимые ею вопросы ответить можно отрицательно. Важные для работы в режиме реального времени параметры установим потом, при посредстве

```
make menuconfig
```

В появившемся текстовом диалоге следует пробежаться по некоторым разделам конфигурационного меню:

» В **Enable the block layer > IO Schedulers > Default I/O scheduler** нужно выбрать вариант **Deadline** (см. LXF126/127).

» В **Processor type and features** необходимо включить поддержку таймеров высокого разрешения (**High Resolution Timer Support**) и установить режим вытеснения (**Preemption Mode**) в **Complete Preemption (Real-Time)**, а для частоты прерываний таймера (**Timer frequency**) указать **1000 Hz**.

» В **Device Drivers > Character devices** активируйте устройства **Enhanced Real Time Clock Support (legacy PC RTC driver)** и **HPET – High Precision Event Timer**, а затем переместитесь в ветку **Sound card support > Advanced Linux Sound Architecture** и включите опции **RTC Timer support** и **Use RTC as default sequencer timer**.

» Наконец, отключите отладку ядра в **Kernel hacking > Kernel debugging**: в нашем случае это пустая трата ресурсов.

Далее собираем и устанавливаем ядро обычным путём:

```
make && make modules_install
cp arch/i386/boot/bzImage /boot/kernel-2.6.33.1-rt11
cp System.map /boot/System.map-2.6.33.1-rt11
```

после чего настраиваем загрузчик – в зависимости от того, какой установлен. В двоичных дистрибутивах более «правильным» путём будет сборка и последующая установка пакета с ядром, но это уже выходит за рамки данной статьи.

Реальное время пошло

Вот теперь можно перезагружаться и проверять работоспособность RT-ядра. Если загрузка прошла успешно, надо запустить *rtirq* – пока сделаем это вручную, выполнив от имени суперпользователя команду

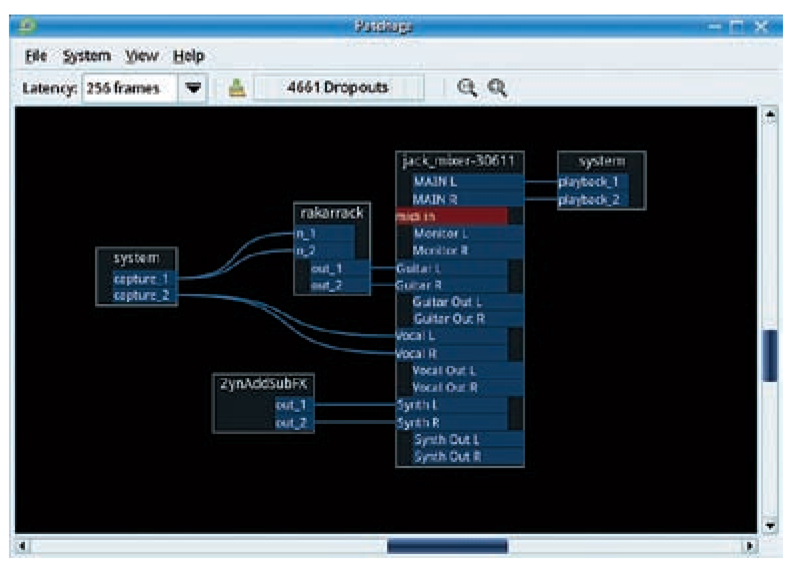
```
/etc/rc.d/init.d/rtirq start
```

В отчёте можно будет наблюдать примерно такую картину:

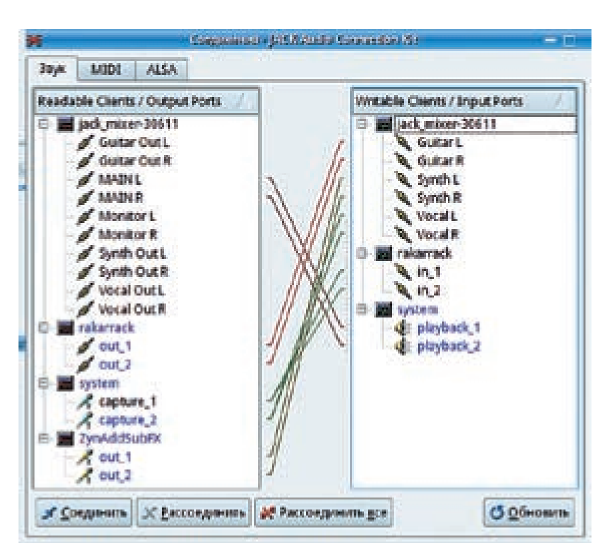
```
Setting IRQ priorities: start [rtc] irq=8 pid=1282 prio=90: OK.
Setting IRQ priorities: start [HDA NVidia] irq=22 pid=1966 prio=85:
OK.
Setting IRQ priorities: start [E-mu 0404b PCI [MAEM8852] (rev.0,
serial:0x40021102)] irq=19 pid=1945 prio=84: OK.
Setting IRQ priorities: start [ohci_hcd] irq=22 pid=954 prio=80:
OK.
Setting IRQ priorities: start [ehci_hcd] irq=23 pid=951 prio=80:
OK.
```

»

› Все параметры запуска JACK можно настроить с помощью *QjackCtl*: он даже покажет время задержки звука.



› В *Patchage* все соединения показаны очень наглядно. Можно подключить ещё немало программ, и всё равно разобраться в том, что делается со звуком, удастся без труда.



› Здесь показана та же схема подключения, но в представлении *QjackCtl*. Понять что-либо невозможно, а ведь неплохо бы добавить по крайней мере обработку вокала и программу записи...

```
Setting IRQ priorities: start [i8042] irq=1 pid=920 prio=75: OK.
```

Если тут всё в порядке, то с чистой совестью добавим сценарий на уровень запуска по умолчанию. Как правило, это уровень 5, но не помешает всё же проверить:

```
grep :initdefault: /etc/inittab
```

Если в выводе будет

```
id:5:initdefault:
```

значит, используется 5-й уровень. Выполняем от имени суперпользователя команду

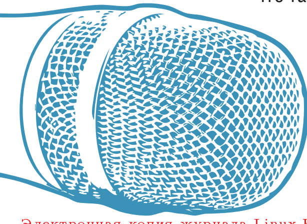
```
chkconfig --add rtirq --level 5
```

Теперь всё готово для запуска JACK, и можно, наконец, его опробовать. На первый раз достаточно запустить *QjackCtl* и нажать на кнопку загрузки сервера. Если сервер запустился успешно и индикатор RT мигает, то всё в порядке. Для проверки подойдёт любой аудиопроигрыватель с возможностью вывода через JACK. Наслушавшись музыки, остановим сервер и перейдём к его более тонкой настройке.

Если при запуске JACK вы не увидели индикатора «RT», значит, сервер выполнялся не в режиме реального времени. Включить его можно в окне настройки *QjackCtl*. Здесь также находятся параметры, определяющие величину задержки сигнала. Это частота дискретизации, число выборок (фреймов) в буфере и число периодов на буфер. Если точнее, то задержка равна произведению числа выборок на число периодов, делённому на частоту дискретизации. Тянуться за калькулятором не надо: *QjackCtl* сам вычисляет эту величину для облегчения настройки.



Какую частоту дискретизации выбрать – решайте сами, исходя из возможностей звуковой карты и ваших запросов. В большинстве случаев подойдут стандартные 48 000 Гц, хотя в случае, если запись производится в расчёте на будущую обработку, иногда имеет смысл выставить 96 000 Гц, а то и 192 000 Гц. Не стоит только забывать, что увеличение этого значения увеличивает затраты на обработку звука, что также может стать причиной заиканий и хрипов (xruns).



Число периодов на буфер почти всегда можно использовать равное 2; только в случае с USB-аудиокартами рекомендуют выставить значение 3. А вот число выборок в буфере следует подбирать эмпирически, уменьшая до тех пор, пока не появятся помехи (или пока аудиодрайвер не откажется использовать такое низкое значение). Как правило, с числом выборок 128 проблем не бывает.

З-заикаемся-с?

В некоторых случаях добиться качественного звука путём настройки *JACK* не удаётся: он всё равно хрипит, а в выводе сервера появляются сообщения вида

```
18:05:27.832 XRUN callback (1).
```

Это означает, что, скорее всего, звуковая карта настроена не оптимальным образом. Следует проверить, не используется ли назначенное ей прерывание заодно ещё каким-нибудь устройством:

```
cat /proc/interrupts
```

Если прерывание назначено только звуковой карте, как в этом примере

```
19:    194541    841686   IO-APIC-fasteoi   EMU10K1
```

то всё в порядке, а вот если оно используется чем-то ещё, как здесь –

```
22:   317650    220908   IO-APIC-fasteoi   ohci_hcd:usb2, HDA Intel
```

то надо исправлять ситуацию.

Возможных решений два. Первое – простое, но подойдёт не для всех компьютеров: нужно вручную назначить IRQ для аудиокарты в настройках BIOS. К сожалению, сделать это можно только на немногих материнских платах. Второе решение потребует взять в руки отвёртку: если назначить IRQ напрямую нельзя, следует просто переставить карту в другой слот. После этого нужно загрузить систему и вновь проверить список прерываний, назначенных устройствам.

Даже назначение уникального IRQ не всегда помогает добиться успеха. Дело в том, что аудиокарта – не единственное устройство на шине PCI, и передача данных прочей периферии влияет на ее работу. Для регулирования этого процесса для каждого устройства устанавливается таймер задержки. Он определяет, какое время после окончания передачи устройство продолжает занимать шину. Соответственно, при работе со звуком оптимальным является вариант, когда задержка для звуковой карты максимальна, а для прочих устройств – минимальна. По умолчанию это не всегда так; для проверки следует изучить вывод команды

```
lspci -v
```

Вот пример настроек аудиокарты моего компьютера:

```
01:07.0 Multimedia audio controller: Creative Labs SB0400 Audigy2 Value
        Subsystem: Creative Labs Device 4002
        Flags: bus master, medium devsel, latency 64, IRQ 19
        I/O ports at dc00 [size=64]
        Capabilities: [dc] Power Management version 2
        Kernel driver in use: EMU10K1_Audigy
        Kernel modules: snd-emu10k1
```

Слова «latency 64» в третьей строке означают, что значение таймера задержки установлено в 64. Для прочих устройств оно равно 0, и проблем при этом не наблюдается. Изменить эти значения можно, выполнив от имени суперпользователя команду

```
setpci -v -s *:* latency_timer=0
```

Это установит задержки для всех устройств в 0. Увеличить же задержку для аудиокарты можно командой

```
setpci -v -s 01:07.0 latency_timer=40
```

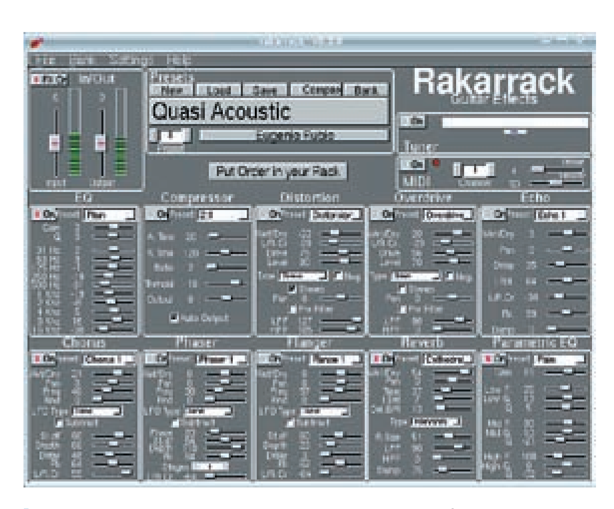
Число здесь вводится в шестнадцатеричной системе, то есть 40 соответствует 64 в десятичной системе. Адрес устройства (в данном примере 01:07.0), конечно, нужно заменить на ваш, из вывода **lspci -v**.

Приведённые здесь числа указаны только для примера. В каждом случае величины таймера задержки следует подбирать индивидуально (в диапазоне от 0 до 255), общее правило заключается лишь в том, что для аудиокарты оно должно быть самым высоким. Когда подходящие значения будут подобраны, их можно установить в BIOS, чтобы не писать скрипт для изменения настроек после каждой перезагрузки.

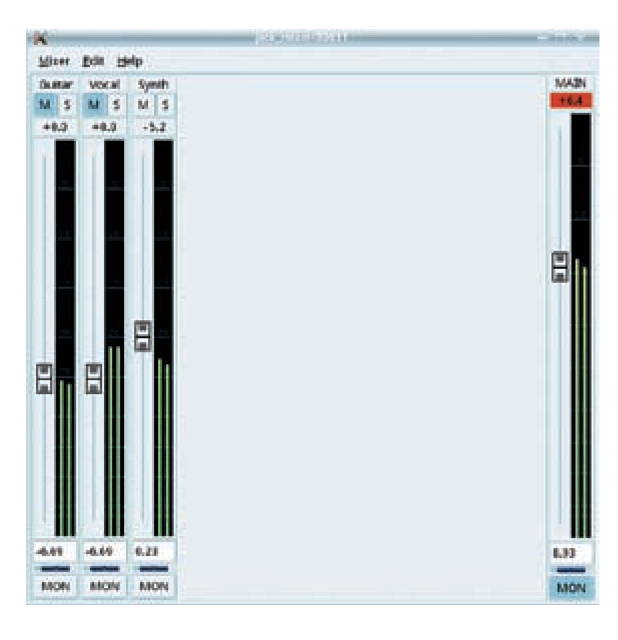
Микшерный пульт

Помимо малой задержки звука, основное преимущество, которое даёт JACK – это возможность легко соединять программы для аудиообработки, точно так же, как аппаратные устройства соединяются последовательно кабелями. Делать это можно и в *QjackCtl*, и непосредственно во многих клиентах JACK. Самое сложное тут – не запутаться в виртуальных «проводах», поэтому для подключения лучше всего пользоваться программой *Patchage*. В ней всё выглядит наиболее наглядно.

Какое ПО понадобится для обработки звука, зависит от конкретной ситуации. Однако есть джентльменский набор программ, держать которые на компьютере стоит всегда. В первую очередь это *JAMin*, сочетающая в себе компрессор и эквалайзер (аж 1023-полосный!). Подключение его на выходе сигнала будет очень полезным. Работу с *JAMin* освещал на страницах **LXF** создатель 64 Studio Дэниел Джеймс [Daniel James]. Это было в **LXF105**, и вы можете найти статью на вики *Linux Format* по адресу: http://bit.ly/bg8N0C.



› 10 эффектов Rackarrack удовлетворят запросы большинства гитаристов, хотя большая их часть — от эквалайзера до ревербератора — будут полезны и прочим музыкантам.



› Микшер должен быть простым и легко управляемым, и *jack_mixer* этим требованиям вполне удовлетворяет.

Гитаристам должен понравиться виртуальный гитарный процессор *Rackarrack*: он сочетает до десяти эффектов в любой последовательности, имеет 80 общих пресетов и по несколько пресетов для каждого эффекта. Впрочем, пригодится он не только для гитары: в нём есть и эффекты общего назначения.

В начале статьи упоминалось о многочисленных модулях LADSPA, позволяющих вытворять со звуком практически всё что угодно. Самый простой способ задействовать любые из них – воспользоваться *JACK Rack*. Эта программа служит для подключения неограниченного числа модулей (ладно, не будем привирать – оно ограничено числом портов JACK) и их настройки.

Ну и, конечно, никак не обойтись без микшера. Тут выбор довольно широк; например, можно установить *jack_mixer*. Любое число моно- и стереовходов и выходов, отдельный выход на мониторы – что ещё надо для счастья?

Проблемы с программами

Не все программы для обработки звука ведут себя в связке с JACK одинаково хорошо: связь с ними может «теряться», а сами они – служить причиной хрипов, при том, что сервер работает нормально.

Причиной «потери» программы сервером может быть слишком малый тайм-аут соединения, установленный по умолчанию. При возникновении проблем следует увеличить это значение в настройках *QjackCtl* – например, до 1000 мс или, для верности, 2000 мс. Как правило, этого достаточно, чтобы подобное не повторялось, но от проблемы с хрипами не избавляет.

Бороться с последними можно путём запуска программ с высоким приоритетом. Тут, однако, следует быть весьма осторожным: если программа работает нестабильно, это может приводить к зависаниям всей системы. Даже в случае со стабильными программами перестараться – значит, навредить: приоритет, более высокий, чем у *JACK* или потоков, работающих с прерываниями аудиоустройств, скорее всего, только ухудшит ситуацию. Приоритет 51 – это вполне разумное значение.

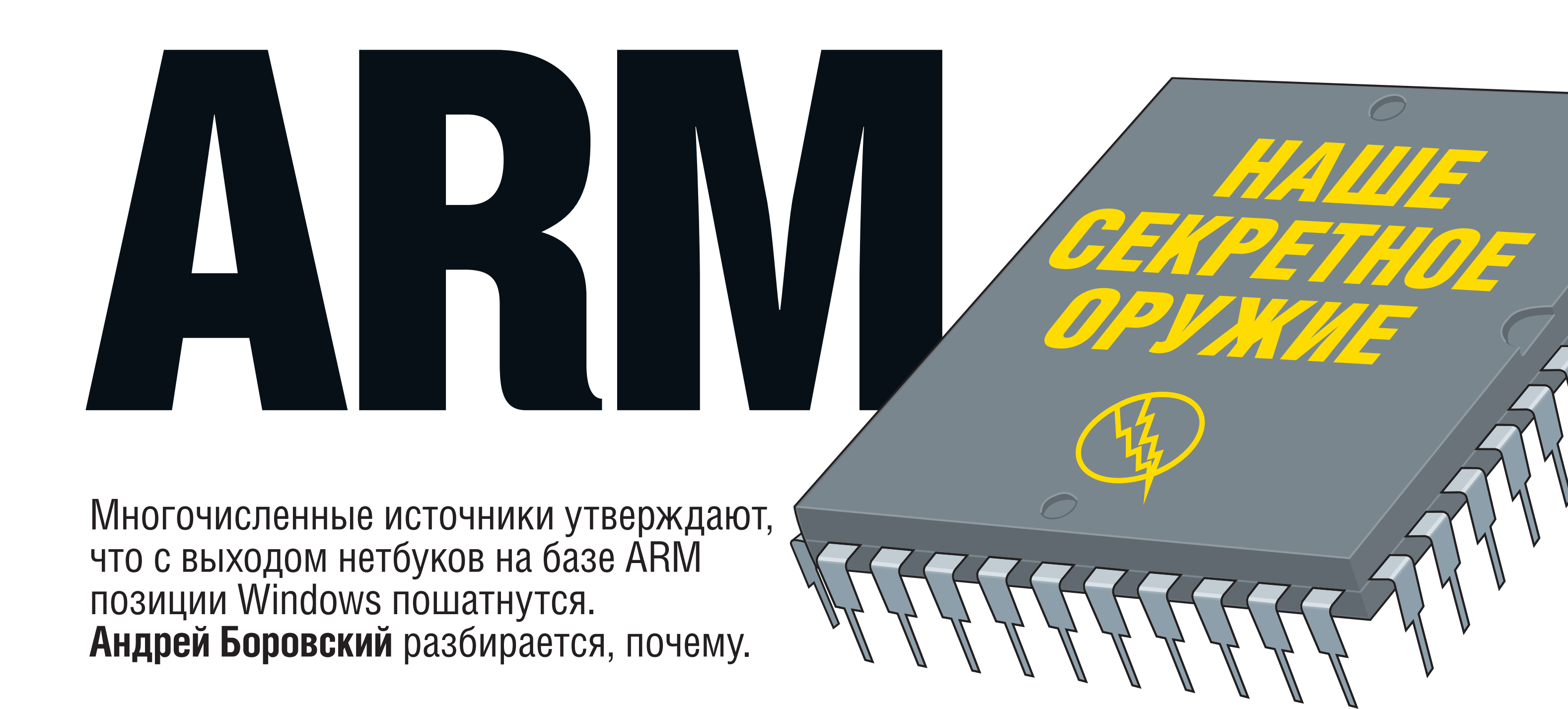
Чтобы запустить программу, например *JAMin*, с повышенным приоритетом, следует выполнить команду

```
chrt 51 jamin
```

В результате количество xrun'ов заметно уменьшается, либо они пропадают вовсе – правда, нагрузка на процессор существенно возрастёт. **LXF**



ARM



Многочисленные источники утверждают, что с выходом нетбуков на базе ARM позиции Windows пошатнутся. **Андрей Боровский** разбирается, почему.



Наш эксперт

Андрей Боровский провел этот месяц, пытаясь найти способ импортировать Iyonix PC на территорию Российской Федерации.

Мир высоких технологий напоминает Северную Америку начала XIX века: помимо хорошо обжитых территорий, где цивилизация уже приняла устоявшиеся формы, есть Дикий Запад, который по-прежнему нуждается в проводниках, исследователях и отважных авантюристах. Мобильные устройства – один из сравнительно новых и все еще малоосвоенных уголков ИТ-мира. Карманные телефоны и компьютеры нельзя назвать новинкой, но в последние несколько лет появились устройства, сочетающие в себе мощь ПК начала этого века, широкополосные каналы передачи данных и мобильность сотового телефона. В этой связи нельзя не отметить удивительную гибкость Linux. Кто бы мог представить, что ОС, корни которой лежат в мире больших многопользовательских машин, так хорошо подойдет для карманных компьютеров?

> «Разработчики ARM могли позволить себе простой RISC-дизайн.»

Мобильные устройства даровали вторую жизнь и процессорам семейства ARM, о которых в последнее время трезвонят все СМИ. Их главное преимущество перед Intel – способность показывать высокую производительность при низкой тактовой частоте (а значит, и энергопотреблении). Не так давно компания ARM распространила видеоролик, в котором продемонстрировала сравнение производительности процессоров Cortex-A9 и Intel Atom: результаты были сопоставимы, а частота работы ARM9 – меньше в три раза. В то время как появление каждого нового процессора Intel сопровождает изрядная шумиха, процессоры ARM тихо делают свое доброе дело, и вы, скорее всего, уже пользуетесь ими, даже не подозревая об этом. Маршрутизаторы Marvell и смартфоны Nokia, Google Nexus One и IP-телефон Snom-100 базируются на ARM и Linux.

Эта статья задумана как туристический путеводитель по стране мобильного ARM: мы сделаем обзор открытого ПО, популярного оборудования и средств разработки. Linux и его инструменты обычно кросс-платформенны, и производители мобильной техники получают целую экосистему программ и утилит даром. Тем не менее, создание приложений для мобильных устройств обладает своей спецификой, особенно когда система разработчика и целевая система используют разные архитектуры.

Что такое RISC?

Ранние микропроцессоры были ориентированы на разработку программ на языке ассемблера (или даже в машинных кодах). Для них было характерно наличие большого числа инструкций, выполняющих сложные операции. Поскольку регистры в прямом смысле стоили дорого, их было немного, а операнды практически всех команд могли поступать как из регистров, так и из ОЗУ.

По мере развития компиляторов и других технологий разработчикам стало очевидно, что более эффективна другая архитектура – со сравнительно небольшим числом инструкций и с большим объемом регистрового файла. Она получила название RISC – Reduced Instruction Set Computer (Компьютер с сокращенным набором команд), а прежняя архитектура была названа CISC (Complete Instuction Set Computer). Практически все современные процессоры (за исключением Intel) относятся к архитектуре RISC.

История героя

Судьба – ироничная штука. Семейство Intel x86 ведет свой род от процессоров, которые предназначались для микроконтроллеров, а ARM изначально задумывался как ЦП для персональных настольных компьютеров (и даже «работал по специальности» в ПК серии Acorn Archimedes и ее немногочисленных клонах) – но увы, к моменту выхода первого ARM (1985 год) лидер в мире ПК уже определился. И хотя ARM1 – настоящий 32-битный процессор с тактовой частотой 8 Мгц – мог померяться силами с тогдашним фаворитом, 80286 (80386 появился в том же 1985 году, но на рынок ПК вышел значительно позже), в дальнейшем ARM'ам пришлось осваивать устройства поскромнее.

Одним из преимуществ ARM перед конкурентами было то, что их создатели могли начать жизнь с чистого листа. Если инженерам Intel приходилось решать проблемы обратной совместимости, разработчики ARM могли позволить себе простой и элегантный RISC-дизайн. Мудрые решения принесли достойные плоды: если в сегменте ПК ARM не повезло, то в других областях, где требовались производительные и экономичные процессоры, эта архитектура оказалась чрезвычайно популярной. Считается, что чипов

ARM было произведено больше, чем любых других. Как и ранняя Intel, компания ARM не стремилась монополизировать производство процессоров своей архитектуры, так что большая их часть изготовлена за пределами ARM. Среди лицензиатов отметились Apple, Broadcom, легендарная Digital Equipment Corporation и сама Intel с XScale.

Неудивительно, что при таком многообразии производителей история поколений ARM выглядит немного запутанной. Помимо названий самих процессоров (ARM1, ARM7TDMI, ARM11, ARM Cortex), в технических спецификациях принято указывать версию архитектуры (например, ARMv1, ARMv4T, ARMv7).

Современная история ARM начинается с архитектуры ARMv4. В ней был впервые реализован блок управления памятью, что позволило запускать современные ОС, в том числе Linux. Надо заметить, что далеко не все области применения ARM требуют защиты памяти и многозадачности, поэтому разработка и выпуск процессоров без блока управления памятью продолжаются и в наше время.

Архитектура ARMv4 и новейшая ARMv7 различаются весьма значительно. Если в первой отсутствовала операция целочисленного деления (не говоря уже о делении с плавающей точкой), то в новых редакциях предусмотрены команды, предназначенные для ускорения обработки мультимедиа-данных (аналог наборов Intel SSE), что, вероятно, нарушает принципы идеологии RISC (см. врезку), но способствует распространению процессоров ARM на современных мобильных устройствах. Полное описание поколений ARM (как и другие полезные сведения) вы найдете в книге «ARM System Developer's Guide» (ISBN 1-55860-874-5), выпущенной Elsevier (см. http://books.google.com/books?id=vdk4ZGRqMskC).

ARM: Особенности архитектуры

Как и положено RISC-системе, ARM обладает расширенным набором регистров. Всего их 17, причем 13 из них являются регистрами общего назначения (т.е. могут использоваться по усмотрению программиста).

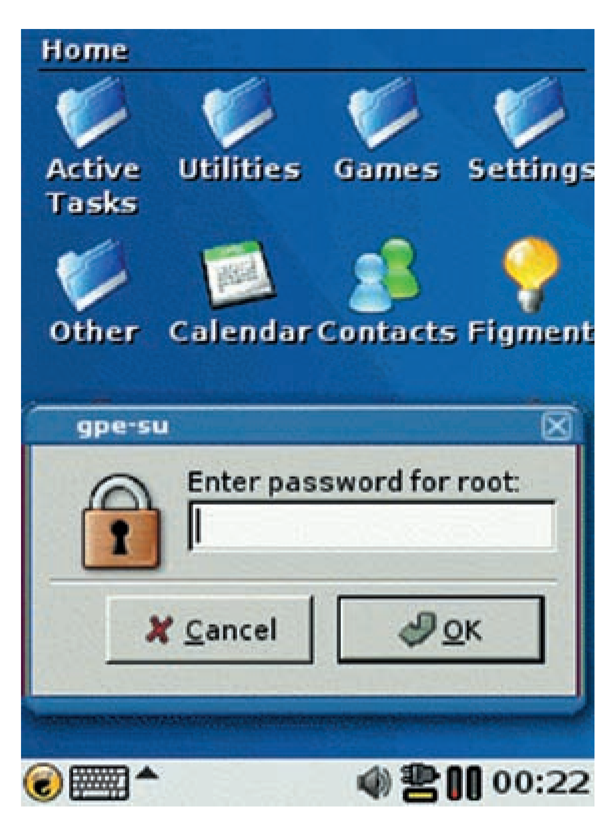
Адрес возврата из процедуры сохраняется не в стеке, а в специальном регистре. Это позволяет реализовать над ним весьма элегантные операции, но нужно помнить, что вложенных вызовов процедур бывает много, а регистр возврата – только один. Перед вызовом очередной процедуры его содержимое нужно где-то сохранять (а затем – восстанавливать).

Большая часть инструкций ARM включает префикс условного выполнения, позволяющий организовать ветвление без операций перехода. Команда выполняется лишь тогда, когда состояние флагов, установленных последней операцией сравнения, отвечает условию префикса. При грамотном использовании это серьезно повышает производительность.

Историю развития Intel можно описать как переход 8-16-32-64 бита, а у ARM наблюдалось обратное движение: изначально 32-битные (кстати, при этом все инструкции одинаковы по длине – 4 байта), данные процессоры обзавелись добавочным 16-битным режимом, с целью повысить компактность кода и быстродействие на 16-битных шинах. Регистры процессора остаются 32-битными, но не все они доступны так же свободно, как в 32-битном режиме. Порядок байтов процессора в ARM может быть как прямым (little-endian), так и обратным (big-endian).

Linux на мобильном ARM

Дистрибутивов Linux для ARM существует немало: практически у каждого разработчика мобильных устройств с поддержкой Linux есть собственный вариант, не считая инициатив сообщества: например, портирования Linux на Windows-смартфоны HTC. Мы же рассмотрим две системы, ориентированные сразу на несколько устройств (они также могут служить основой для дальнейших разработок).



› Так выглядит ARM Linux Internet Platform.

» ARM Linux Internet Platform (linux.onarm.com)

Данный продукт создается группой заинтересованных компаний специально для мобильных устройств. ARM Linux Internet Platform – это своего рода Moblin (простите, MeeGo) для ARM. Логически система разделяется на три уровня. Самый нижний – Linux Board Support Package – включает ядро с драйверами, стандартные библиотеки C, оптимизированные под конкретную модель процессора, и средства настройки оборудования. Поверх него реализована общая Linux-платформа (Linux Generic Platform), включающая шину D-Bus, слой абстракции *HAL* от FreeDesktop.org, виртуальную файловую систему *GnomeVFS*, *X*-сервер, менеджер сетей *NetworkManager* и многое другое. На самом верхнем уровне реализован пакет приложений (Application Suite), содержащий компоненты графического интерфейса пользователя, движок для просмотра web-страниц, мультимедиа. В качестве оконного менеджера используется *Matchbox*, известный пользователям Linux-устройств Openmoko Neo 1973 и Nokia 770.

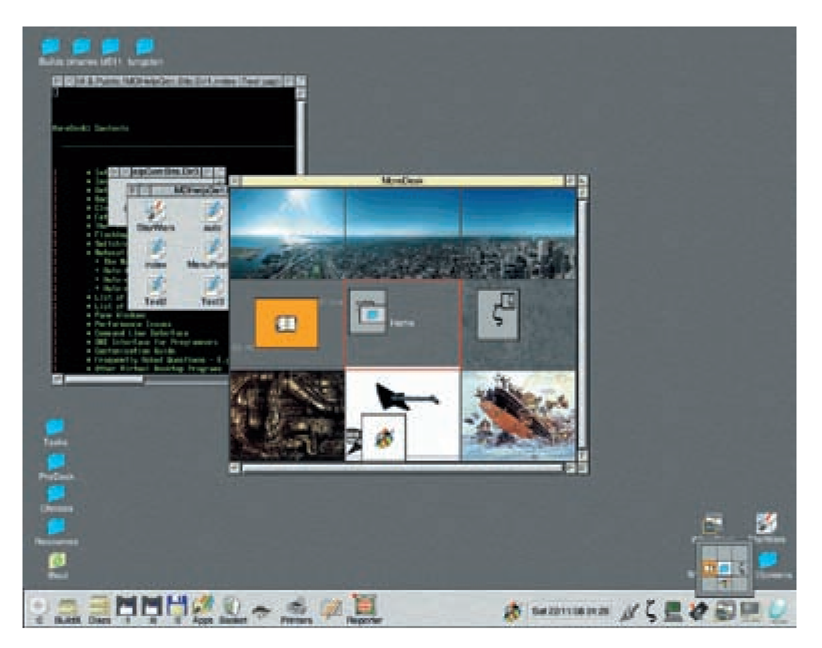
Для разработки приложений можно применять основанный на *GCC* инструментарий *Scratchbox* (www.scratchbox.org), отличительной чертой которого является возможность не только скомпилировать программу для ARM на платформе x86, но и сразу же запустить ее на выполнение в *Qemu* (подробности ниже).

В настоящее время на основе Linux Internet Platform реализовано несколько готовых систем. Это базовая платформа (Generic Platform), задуманная, как нетрудно догадаться, для последующей специализации. По умолчанию она ориентирована на архитектуру ARMv5T и не содержит средств работы со специализированными устройствами. В пакет приложений базовой платформы входят миниатюрный рабочий стол *Matchbox2* (и графическая библиотека *Cairo*), браузер *Nitehawk* и мультимедиа-плейер на основе *GStreamer*. На сайте проекта можно найти инструкции по сборке Generic Platform для *Qemu*.

Платформа n8x0 предназначена для планшетов Nokia N770/N8x0. Mistral OMAP35x EVM – система для плат OMAP35x EVM, Beagle board – Linux для плат Beagleboard. S3c64xx – платформа для платы Samsung SMDK6400 и ее родственников. Zoom2 – Linux-платформа для очень интересного устройства Texas Instruments Zoom II.

» Ångström (www.angstrom-distribution.org)

Видели ли вы когда-нибудь мобильное устройство с чистым консольным интерфейсом? Дистрибутив Ångström предоставит вам »



› Графический рабочий стол RISC OS. Где же вы, ребята, были в 95-м году?

такую возможность. Его название намекает на небольшие размеры и скромные требования. По словам разработчиков, Ångström можно запустить на устройстве с 4 МБ флэш-памяти, а в будущем, утверждают они, система сможет работать и на тостере.

Помимо консольного интерфейса, Ångström поддерживает графическую оболочку *GPE Palmtop Environment* и «настоящий» *X11*. Поставляемый в системе набор шрифтов поддерживает кириллицу, латиницу и иврит. В планах разработчиков – портирование *Qtopia*. Список поддерживаемой аппаратуры включает различные модели Sharp Zaurus, HP iPaq, Nokia 770, HTC Universal и ряд менее распространенных устройств.

Другие ОС

Linux – предпочтительная, но не единственная ОС для процессоров ARM. **Symbian, Windows CE, QNX – вот лишь часть сис**тем, рассчитанных на этот процессор. Мы же рассмотрим внимательнее две из сравнительно редко используемых, но весьма интересных ОС.

Было бы странно, если бы Acorn Archimedes, полноценный ПК своего времени, не обладал операционной системой. Она называлась RISC OS (www.riscos.com) и уже в 1988 году поддерживала добровольную корпоративную многозадачность, что продолжает делать и до сих пор (напомню, что в современных ОС применяется принудительная – вытесняющая многозадачность). Если бы звезды расположились иначе, RISC OS вполне могла бы занять нишу MS DOS. Как это ни удивительно, RISC OS до сих пор разрабатывается несколькими компаниями, в том числе Castle Technology. Это однопользовательская система с крайне слабыми средствами защиты программ друг от друга. У нее есть несколько графических оболочек, выполненных на уровне конца 90-х, и набор приложений того же уровня.

› ARM Linux в окне Qemu, запущенном в Linux для x86.

Хотя настоящий Linux может работать только на процессорах с аппаратной защитой памяти, для чипов, лишенных этой роскоши, существует своя Linux-подобная ОС – uClinux (www.uclinux.org), изначально базирующаяся на исходных текстах ядра Linux 2.0. В отличие от RISC OS, uClinux не ориентирована специально на процессоры ARM и поддерживает множество других архитектур.

uClinux сделана максимально похожей на Linux, но из-за принципиальной разницы в используемом оборудовании между этими ОС существуют некоторые отличия. В uClinux отсутствует защита памяти, так что любой процесс может «завалить» всю систему. Все процессы выполняются в общем адресном пространстве, и нет системного вызова **fork()** – есть только функция **vfork()**, которая похожа, но не идентична одноименной в Linux. Поскольку старый и новый процессы разделяют адресное пространство, попытка выполнять их одновременно может привести к ошибкам. В uClinux выполнение процесса-родителя, вызвавшего **vfork()**, приостанавливается до тех пор, пока не завершится дочерний процесс. Вместо классической функции **malloc()** в uClinux используется вызов **mmap()** (библиотека *uClibc* – аналог *libc* – в целях повышения совместимости реализует **malloc()** как обертку вокруг **mmap()**).

ARM на вашем ПК

Переходя от программной части нашего экскурса к аппаратной, нельзя не упомянуть *Qemu*, виртуальную машину, способную эмулировать процессоры ARM. Не побоюсь сказать, что каждый разработчик программ для ARM Linux имеет дело с *Qemu*. Дело в том, что его можно использовать не только для запуска ARM-программ, но и для их отладки (именно так и поступает уже упомянутый *Scratchbox*).

Поскольку *Qemu* входит во все популярные дистрибутивы Linux и собирать его из исходных текстов, в общем-то, нужды нет, мы опустим подробности данной процедуры. Вместо этого рассмотрим установку самого Linux в виртуальную машину. Вам понадобится либо дистрибутив Linux для ARM в виде образа, понятного *Qemu*, либо образ диска *Qemu* с уже установленным ARM Linux. Образы ядра и временной файловой системы можно н айти, например, на http://people.debian.org/~aurel32/arm-versatile/, а образы дисков с предустановленным Debian – на http://people.debian.org/~aurel32/qemu/arm/. На странице http://www.oszoo.org/wiki/index.php/Category:ARM_images приведен torrent для образов покрупнее (скажем, с LXDE), а если вы предпочитаете не Debian, а Fedora, обратитесь к ftp://ftp.linux.org.uk/pub/linux/arm/fedora/qemu/ и http://cdot.senecac.on.ca/arm/.

Если вы хотите установить Linux на *Qemu* самостоятельно, сначала нужно создать образ диска. Например, команда

```
qemu-img create -f qcow hda1.img 8G
```

сгенерирует его в формате *Qemu* и объемом 8 ГБ. Далее вам понадобятся ядро системы (например, **vmlinuz-2.6.18-6-versatile**) и образ временной файловой системы (скажем, **initrd.img-2.6.18-6-versatile**). При использовании названных компонентов команда для запуска *Qemu* в режиме эмуляции ARM может выглядеть так:

```
qemu-system-arm -M versatilepb -kernel vmlinuz-2.6.18-6-
versatile -initrd initrd.gz -hda1 hda.img -append "root=/dev/ram"
```

В результате будет начата стандартная процедура инсталляции Linux. При запуске системы с образа, на котором она предустановлена, команда для старта *Qemu* может выглядеть вот так:

```
qemu-system-arm -M versatilepb -kernel zImage-versatile -hdc
rootfs-f10-dev
```

Добавление поддержки сети в Qemu – отдельная тема. Скорее всего, виртуальный сетевой адаптер будет единственным средством обмена данными между вашей ARM-машиной и внешним миром. Инструкции по настройке виртуальной сети для *Qemu* зависят от параметров конкретной ОС, так что рекомендуем вам обратиться к документации и форумам.

Оборудование

Как уже было сказано, в мире существует великое множество устройств, сердцем (или мозгом?) которых служит ARM. Ниже приведены три из них, причем каждое (в некотором роде) характеризует целый сектор применения ARM и Linux.

» TI OMAP Zoom II

Если вы ничего не слышали о Texas Instruments OMAP34X Zoom II, в этом нет ничего удивительного. Данное устройство рассчитано не на конечных пользователей, а на разработчиков мобильных приложений. Тем не менее, возможности Zoom II способны порадовать сердце любого ценителя гаджетов: 4-дюймовый емкостной сенсорный экран с поддержкой модной нынче технологии жестов и HD-видео (устройство может выводить видео и на большой HD-экран), настоящая полноразмерная клавиатура, Wi-Fi, Bluetooth, поддержка 3G-модема, возможность установки SIM-карты, 8-МПикс камера и акселерометр, процессор TI OMAP3430, основанный на новейшей архитектуре ARMv7, 256 МБ быстрой DDR-памяти и 512 МБ флэш – очень достойные характеристики.

Аппаратная поддержка OpenGL ES 2.0 (ответ платформе Intel Atom) позволяет устройству работать с 3D-графикой в реальном времени, а всевозможные расширенные наборы инструкций ускоряют общую обработку мультимедиа. К этому надо добавить входящую в комплект мощную отладочную плату (одних только разъемов mini-USB – 5 штук!) и средства для разработки приложений. Symbian Foundation использует TI OMAP Zoom II как эталонное устройство при создании открытых приложений Symbian. Zoom II не обошли вниманием и разработчики Android и мобильных Linux-систем. Впрочем, если вам уже не терпится заполучить Zoom II в собственное пользование, мне, возможно, придется охладить ваш пыл: его цена существенно превышает 1000 долларов.

> «Существует множество устройств, сердцем которых служит ARM.»

» Iyonix PC

Если же вы – любитель пооригинальничать и хотите иметь у себя дома ПК с процессором ARM, вам не обязательно добывать где-то «старину Архимеда». Компания Castle Technology, одна из наследниц Acorn, предлагает всем желающим Iyonix PC, основанный на процессоре Intel XScale и оснащенный новой версией RISC OS, но поддерживаемый и распространенными дистрибутивами ARM Linux для настольных ПК. По отзывам пользователей, производительность Iyonix PC сопоставима с ранними Pentium IV, в то время как цена соответствует современным компьютерам сегмента Low-End. Что же касается возможности приобретения Iyonix PC в нашей стране, то я предоставляю желающим (буде таковые найдутся) пройти этот квест самостоятельно.



› TI OMAP34X Zoom II — мечта хакера.



› HP iPAQ собственной персоной.

» HP iPAQ

Если вы собираетесь заняться программированием для ARM, эмулятор *Qemu*, при всех своих возможностях, вряд ли сильно вдохновит вас. В конце концов, мы ведь хотим видеть наши программы на настоящем устройстве, не так ли? Однако профессиональное решение, подобное TI OMAP Zoom 2, для большинства из нас будет слишком дорогим, а Iyonix PC – слишком экзотичным (да и доставить в нашу страну этот ПК будет непросто). Я уж не говорю о том, что при всей нашей любви к электронной технике ради нее самой, далеко не все линуксоиды захотят покупать устройство, которое трудно применять для чего-то еще, кроме отладки собственных приложений.

Хорошим выбором для Linux-разработчика могут стать наладонники и смартфоны iPAQ, которые ныне выпускаются компанией Hewlett-Packard. Одно из преимуществ HP iPAQ – активная поддержка со стороны сообщества Linux, в частности, разработчиков *Scratchbox*. Устройства продаются с предустановленной мобильной Windows, но установить на них Linux – не проблема. На сайте www.handhelds.org (и многих других) можно найти советы, как оснастить iPAQ различными дистрибутивами (как минимум – Familiar Linux и Ångström), создав резервную копию Windows. После базовой установки Linux вы сможете подключить устройство к внешнему миру, используя стек TCP/IP, через Wi-Fi, Bluetooth, USB и даже последовательный порт. Теперь вы получаете всю ту свободу, которую обычно дает вам Linux. Можно поставить инструментарий разработчика GNU на само устройство, но удобнее все же установить кросс-платформенную систему сборки на «большом компьютере».

Итак, будем считать, что вы выбрали платформу для тестирования и отладки программного обеспечения, ориентированного на ARM. В следующей статье речь пойдет о самих средствах разработки. LXF

Zsh обустройте свою оболочку



Одев гидрокостюм и отправившись в морские глубины, **Коэн Вервлоесем** враз осознал всю важность удобной оболочки.

Многие пользователи никогда не меняют оболочку, установленную по умолчанию в их дистрибутиве Linux, и почти всегда этой оболочкой является *Bash* (сокращение от *Bourne-again shell*). Это неплохо, если вы не используете расширенные возможности, но если вы много работаете с командной строкой, не помешает присмотреться и к другим оболочкам, а не только потреблять что дают. Оболочка – важное звено между вами и вашей операционной системой. Если она правильно настроена, ваша работа станет удобнее, эффективнее и производительнее.

> «Zsh обладает многими преимуществами Bash и сверх того.»

В данной статье речь пойдет о *Zsh* (*Z shell*) – оболочке, обладающей многими преимуществами *Bash* и других оболочек, а также массой возможностей сверх того. После прочтения этой статьи вы получите представление о мощи *Zsh* и сможете принять информированное решение о том, стоит ли переключаться на нее со стандартной оболочки, предлагаемой вашим дистрибутивом. Основное внимание мы уделим интерактивному использованию *Zsh*, в меньшей степени – неинтерактивным сценариям, то есть сконцентрируемся на ежедневной работе, а не на скриптописании и автоматизации.

Удобной возможностью многих оболочек является подстановка [globbing], известная также как генерация имен файлов. Широко известный пример – ввод имени ***.txt**, которое подразумевает все файлы, заканчивающиеся расширением **.txt**. Ваша оболочка превратит это в список всех таких файлов и передаст его команде – как, например, в **ls *.txt**.

В *Bash* возможности подстановки довольно ограничены. Например, как быть, если вы хотите перечислить все файлы с расширением **.txt** в подкаталогах текущего каталога? Конечно, можно

Стартовые файлы

Каждая оболочка имеет ряд файлов, используемых ею при запуске для самонастройки. Файлы настройки *Zsh*, действующие в масштабах всей системы, находятся в каталоге **/etc/** (в Ubuntu и ряде других аналогичных дистрибутивов – в **/etc/zsh/**), а настройки для конкретных пользователей располагаются в их домашних каталогах. При запуске *Zsh* оболочка считывает следующие файлы в таком порядке:

- **/etc/zshenv** и **~/.zshenv**
- если оболочка запускается при входе пользователя в систему: **/etc/zprofile** и **~/.zprofile**
- если это интерактивная оболочка: **/etc/zshrc** и **~/.zshrc**
- если оболочка запускается при входе пользователя в систему: **/etc/zlogin** и **~/.zlogin**

А когда пользователь покидает оболочку, запускающуюся при входе в систему [login shell], *Zsh* читает файлы **/etc/zlogout** и **~/.zlogout**.

Работая с командами, которые записываются в каждый из файлов настройки, важно различать типы оболочек. Регистрационная оболочка [login shell] – это экземпляр оболочки, запускаемый при входе пользователя в систему: например, через SSH или на виртуальном терминале.

Интерактивная оболочка отображает для пользователя приглашение к вводу команд – например, если вы в Ubuntu открываете окно Терминал [Terminal]. Однако если вы запустите команду **ssh host some-command**, то это будет регистрационная оболочка, а не интерактивная.

написать **ls */*.txt**, но тогда выведутся **.txt**-файлы только с данного уровня вложенности каталогов. *Zsh* имеет мощное решение по рекурсивному поиску файлов: **ls **/*.txt**. Эта команда выведет все **.txt**-файлы во всех подкаталогах. Оцените мощь этого решения: оно почти что делает команду **find** избыточной. Например, вместо **find . -name *.txt | grep foo**, *Zsh* позволяет использовать куда более простую команду: **grep foo **/*.txt**. Если вы хотите, чтобы *Zsh* отслеживала символьные ссылки, можно воспользоваться конструкцией *****/**; но остерегайтесь бесконечных циклов.

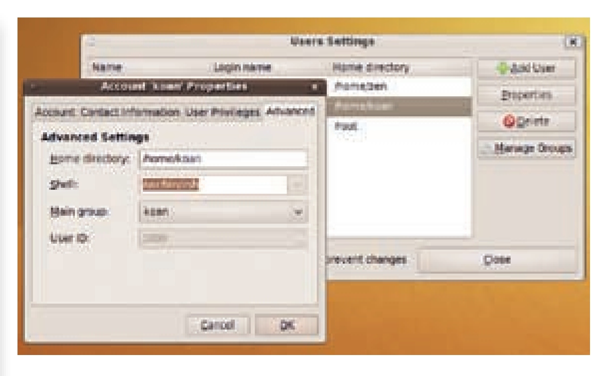
Помимо этого, *Zsh* воспринимает конструкцию **[...]**, подбирающую соответствие любым символам, заключенным в скобки. Например, конструкция ***.[cho]** обозначает все файлы, в расширения которых входят символы **c**, **h** или **o**. Конструкциям **[^...]** и **[!...]** *Zsh* ставит в соответствие все символы, кроме заключенных в скобках.

Использование квалификаторов

Квалификаторы подстановки [glob qualifiers] – еще одно приятное дополнение *Zsh*: эта оболочка умеет выбирать типы файлов по использованию определенных флагов в скобках в конце подстановочного шаблона. Например, чтобы перечислить только каталоги, вложенные в текущий, можно воспользоваться командой:

```
print *(/)
```

Для обозначения только обычных файлов пользуйтесь конструкцией **(.)**; для каталогов – **(/)**, исполняемых файлов – **(*)**, символьных ссылок – **(@)**, сокетов – **(=)**, именованных каналов – **(p)**,



файлов устройств – **(%)**, блочных файлов – **(%b)** и символьных файлов – **(%c)**.

Точно так же можно проверять права доступа к файлам: **(r)**, **(w)** и **(x)** означают файлы, допускающие чтение, запись и выполнение своим владельцем. Сокращения **(R)**, **(W)** и **(X)** выбирают файлы со «всемирными» правами доступа [world permissions], а конструкции **(A)**, **(I)** и **(E)** задают групповые права. Например, чтобы найти все исполняемые файлы в текущем дереве каталогов, скомандуйте

```
ls **/*(.x)
```

Если вы предпочитаете использовать символьные аргументы, распознаваемые командой **chmod**, вы можете сделать и так:

```
print *(f:gu+rw,o-rwx:)
```

Еще один полезный подстановочный квалификатор проверяет, является ли пользователь или группа владельцами файла. Чтобы протестировать собственного пользователя или группу, используйте конструкции **(U)** или **(G)**, соответственно. Для остальных пользователей и групп нужно добавлять идентификатор (ID) пользователя или группы к конструкциям **(u)** или **(g)**. Так, **(u0)** позволит вам искать все файлы, принадлежащие root, а **(u1001)** – все файлы, принадлежащие пользователю с user ID, равным 1001. При желании можно использовать и имена, но их надо заключать между символами двоеточия: например, **(u:koan:)** выбирает файлы, принадлежащие пользователю **koan**.

Zsh может выбирать файлы по времени модификации или доступа – для этого используются конструкции **(m)** и **(a)**, соответственно. Искать можно по точному времени или по периодам: до **(-)** или после **(+)** конкретного момента. По умолчанию используются дни, но время можно измерять и в месяцах **(m)**, неделях **(w)**, часах **(h)**, минутах **(m)** или секундах **(s)**.

Например, вот команда, находящая все файлы, к которым вы обращались за последнюю неделю:

```
print **/*(.aw-1)
```

Следующая команда найдет все файлы, модифицированные за последний час:

```
print **/(.m0)
```

Квалификаторы подстановки по размеру файла работают аналогично: **(L)** указывает размер файла, по умолчанию измеряемый в байтах. Помимо байтов, размер файла можно измерять в килобайтах **(k)**, мегабайтах **(m)**, или в блоках по 512 байт **(p)**. Таким образом можно очень быстро найти в текущем каталоге все файлы, размер которых превышает 1 мегабайт:

```
print *(Lm+1)
```

Квалификаторы подстановки можно произвольным образом комбинировать. Получаемые конструкции могут показаться громоздкими, но при знании основных квалификаторов их толкование не составит труда. Например, ***(u0WLk+10m0)** означает «файлы, принадлежащие пользователю root, к которым все имеют доступ с правом записи; размер их больше 10 КБ; модифицировались в течение последнего часа». Перед квалификатором можно использовать символ **^**, который означает его отрицание; а запятая служит оператором ИЛИ – **or**. На man-странице приведена »

Переменные приглашения Zsh

- **%/** Текущий рабочий каталог
- **%~** Краткое обозначение текущего рабочего каталога (например, ~ для домашнего, но сокращениям подвергаются и именованные каталоги)
- **%t** Время в 12-часовом формате
- **%T** Время в 24-часовом формате
- **%*** Время в 24-часовом формате с секундами
- **%n** Имя пользователя
- **%m** Сокращенное имя хоста
- **%M** Полное имя хоста
- **%#** % для обычных пользователей и # – для root
- **%?** Статус последней команды (**0** – успешное выполнение)

» **Если мощь и гибкость *Zsh* вызвали у вас желание познакомиться поближе, укажите в качестве оболочки по умолчанию /usr/bin/zsh.**

Скорая помощь

Сменить на *Zsh* оболочку по умолчанию можно командой **chsh -s /usr/bin/zsh**. Изменение войдет в силу только после выхода из системы.

более подробная информация обо всех квалификаторах подстановки, включая методы сортировки вывода.

В *Zsh* есть и другие способы упростить вашу работу. Каждая оболочка допускает псевдонимы (aliases), так что вы можете создать для популярных команд альтернативные имена, которые проще вводить с клавиатуры. Но *Zsh* способен на гораздо большее.

Определить псевдоним очень просто. Например, если вы часто вводите команду **ls -lh**, можно создать псевдоним **lh** с помощью следующей команды:

```
alias lh="ls -lh"
```

Псевдонимы удобны для пользователей; для интерактивных оболочек задавайте их в файле **zshrc**. Если после определения псевдонима ввести **lh**, оболочка преобразует ваш ввод в **ls -lh**. Это изменение будет действовать и при указании для **lh** аргументов и опций, например: **lh -a foodir/**.

Глобальные псевдонимы

А если вам хочется создать псевдонимы для команд, не являющихся первым словом в командной строке? *Zsh* заботится и об этом: такие псевдонимы называются глобальными. Для создания глобального псевдонима укажите в команде **alias** опцию **-g**, например:

```
alias -g L="|less"
```

Теперь вам будет очень удобно листать страницы вывода другой команды, просто добавив к ней псевдоним **L** – например, **dmesg L**.

Вот еще несколько полезных глобальных псевдонимов:

```
alias -g ...='../..'
alias -g ....='../../..'
alias -g .....='../../../..'
```

Еще один вариант псевдонимов – суффиксы. Вот несколько поучительных примеров:

```
alias -s tex=vim
alias -s pdf=xpdf
alias -s html=w3m
```

Это означает, что если мы введем имя файла, как если бы это была команда, то файл с расширением **.tex** будет открыт редактором *Vim*, файл с расширением **.pdf** – с помощью *xpdf*, а файл с расширением **.html** – с помощью *w3m*.

Псевдонимы можно создавать и для каталогов, хотя команда **alias** для этого не годится. Часть таких псевдонимов в оболочке уже определена: это, например, **~** для вашего домашнего каталога и **~user** – для домашнего каталога пользователя **user**. Зато командой **hash** можно задавать собственные имена для произвольных каталогов, например:

```
hash -d down=~/Desktop/Downloads
```

Теперь вы сможете перейти в этот каталог из любого другого места файловой системы, введя команду **cd ~down**.

Еще одна опция, упрощающая работу с каталогами, хоть и не являющаяся псевдонимом – это **AUTO_CD**. Если в вашем файле **zshrc** вы установите эту опцию командой **setopt AUTO_CD**, а потом введете с клавиатуры нечто, не имеющее аргументов и не являющееся командой, *Zsh* проверит, не является ли введенный вами текст именем каталога. Например, если вы введете **Documents**, *Zsh* поведет себя так, как если бы вы ввели команду **cd Documents**. Переход в родительский каталог тоже упростится: вместо **cd ..** достаточно будет ввести **..**.

Большинство оболочек автоматически завершают имена файлов по нажатию клавиши Tab. В *Bash* вы можете расширить эту опцию до дополнения имен хостов и т.д., установив пакет **bash-completion**. *Zsh* также имеет мощную и полностью программируемую систему дополнений, но по умолчанию она не активирована. Для ее подключения установите в своем файле **zshrc** такие опции:

```
autoload -U compinit
compinit
```

Когда система автодополнения *Zsh* будет активирована, попробуйте набрать несколько типичных Linux-команд, но вместо ввода аргументов нажмите клавишу Tab. Вы наглядно убедитесь в мощи *Zsh* и увидите, что она может добавлять. Например, у *Zsh* хватает интеллекта для того, чтобы дополнять команду **cd** каталогами. Еще один пример – введите команду **tar -xvzf** и нажмите клавишу Tab. *Zsh* отобразит имена файлов в текущем каталоге, имеющих расширение **.tar.gz**. Если вы введете **tar -xvf**, *Zsh* отобразит только имена **.tar**-файлов. Вы можете даже избирательно извлекать конкретные файлы из архива **.tar.gz**, не вводя полный путь.

Привыкнув к системе дополнений, вы обнаружите, что практически перестали печатать такие команды, как **cd**, **ls** и тому подобные, чтобы определить, где вы находитесь и какие файлы там имеются. Вы начнете печатать свои команды, а система дополнения сама найдет для вас нужные файлы. Это относится не только к файлам, но и к переменным оболочки, именам пользователя, следующим за опцией **-user** и именам хостов, и даже к именам пользователя в SSH. Более того, система работает даже с файлами удаленной учетной записи, при условии, что применяются SCP и сервер с публичными ключами.

Большинству пользователей незачем изменять стандартную систему завершений в *Zsh* – она и так достаточно интеллектуальна. Однако при желании вы можете расширить ее, введя собственные правила завершения или изменив стандартные. Давайте рассмотрим эту возможность на примере SSH. По умолчанию, система дополняет имена всех локальных пользователей и все имена хостов, перечисленные в **.ssh/known_hosts**. Но если вам нужно регистрироваться на двух машинах, лучше задать собственное правило завершения, например:

```
zstyle ':completion:*' users-hosts koan@vervloesem.eu
kvervloe@discovery.example.org
```

Это правило удобно тем, что если вы введете с клавиатуры **koan** и нажмете клавишу Tab, *Zsh* уже будет знать, что имя хоста должно быть **vervloesem.eu**, потому что имени другого хоста соответствует другое имя пользователя.

В список автозавершения можно добавить различные цвета, как показано на иллюстрации внизу этой страницы. Мы будем использовать те же цвета, которые отображает **ls** от GNU с опцией **--color**:

```
zmodload -l zsh/complist
zstyle ':completion:*' list-colors ${(s.:.)LS_COLORS}
```

Здесь предполагается, что **LS_COLORS** установлена как переменная окружения. Если в вашем дистрибутиве это не так, запустите команду **dircolors** и скопируйте ее вывод в ваш файл **zshrc**, вставив его перед только что приведенными строками.

Приглашения оболочки

Приглашение оболочки – это, скорее всего, первая вещь, которую вам захочется настроить. Опытные пользователи видят приглашение командной строки тысячи раз в день, и логично подумать о приятности данного зрелища. К счастью, *Zsh* справляется

Скорая помощь

Если вы определили именованные каталоги, можете установить опцию **CDABLEVARS**: она позволит вам использовать команду **cd down** вместо **cd ~down** или (в комбинации с **AUTO_CD**) даже просто **down**.

```
Terminal
File  Edit  View  Terminal  Help
del% ls Downloads/GoogleEarthLinux.bin
aeskey.txt                        openvswitch-0.90.7.tar.gz
android-sdk_r04-linux_86.tgz      proxmox-ve_1.5-4627-3.iso
bt4-final.iso                     Queen\ -\ Don\'t\ stop\ me\ now.mp3
eicar.txt                         ubuntu-9.10-netbook-remix-i386.iso
english.txt                       zsh.txt
GoogleEarthLinux.bin*
```

› **Списки завершения можно «расцветить» яркими красками.**

Зэтимология

Первую версию **Zsh** написал Пол Фолстад [Paul Falstad], будучи студентом Принстонского университета. Когда он искал имя для своей оболочки, его взгляд упал на входное имя преподавателя-ассистента, Zhong Shao: zsh.

с этой задачей блестяще. Приглашение командной строки управляется переменной **PROMPT**. Например, стандартное приглашение *Zsh* выглядит так: **"%m%#"**. Это значит, что в строке приглашения отображается короткое имя хоста, за которым следует значок **%** (для обычных пользователей) или **#** (для root).

Еще одна уникальная черта *Zsh* – поддержка правостороннего приглашения, управляемого переменной **RPROMPT**, с тем же синтаксисом, что и **PROMPT**. Это удобно, если вам не нравится левостороннее приглашение, изменяющееся по длине. Например, чтобы показать имя текущего каталога, достаточно установить **"%~"** в качестве правостороннего приглашения. *Zsh* справляется и с длинными правосторонними приглашениями. Например, пусть ваш текущий каталог имеет длинный путь; как только ввод команды приблизится к правостороннему приглашению, оно исчезнет.

Для приглашений можно определить условные подстроки. Это удобно, если вам хочется, чтобы часть приглашения зависела от результата выражения. Например, вот как настроить *Zsh*, чтобы при успешном выполнении последней команды в приглашении отображался символ **O**, а в случае неудачи – **X**:

```
PROMPT="%(?.O.X)"
```

Добавим цвета

Монохромное приглашение выглядит уныло, а его текст сливается с выводом ваших команд. Если использовать выделение цветом, то на экран будет не просто приятнее смотреть, но и вывод станет выглядеть более структурированным. Использовать выделение цветом в приглашении можно с помощью управляющих последовательностей [escape sequences].

В состав *Zsh* входит функция *colors*. Когда она загружена и выполняется, она определяет ассоциативные массивы **$fg** и **$bg**, используя соответствующие управляющие последовательности для выбранных цветов. Например, **${fg[red]$bg[yellow]}** создает управляющие последовательности для отображения красного текста на желтом фоне. Следует помнить, что управляющие последовательности, которые не перемещают курсор, всегда должны открываться символами **'%{'**, а замыкаться – **'%}'**.

Чтобы загрузить эти цвета, добавьте в ваш файл **zshrc** следующие строки:

```
autoload colors zsh/terminfo
if [[ "$terminfo[colors]" -ge 8 ]]; then
        colors
fi
```

Теперь вернемся к нашему условному выражению с **O** и **X**. Мы можем добавить туда выделение цветом. Если последняя команда завершилась успешно, нужно отобразить зеленый символ **O**, в противном случае – красный **X**:

```
PROMPT="%(?.%{${fg[green]%}O. %{${fg[red]%}
X) %{${fg[default]}%}
```

Zsh позволяет автоматически запускать определенный код при определенных обстоятельствах. Для этого нужно только реализовать некоторые специальные функции. Две наиболее часто используемые – **chpwd()** и **precmd()**. *Zsh* вызывает **chpwd()** при каждом изменении текущего каталога, а **precmd()** выполняется перед тем, как *Zsh* отобразит новое приглашение. Обе функции регулярно используются для отображения текущего каталога в строке заголовка эмулятора терминала. Если, помимо оболочки, вы применяете другие программы, изменяющие строку заголовка эмулятора терминала (например, *Vim*), воспользуйтесь **precmd()** – после отработки другой программы она восстановит строку заголовка. Отображение текущего каталога в строке заголовка делается так (адаптировано с man-страницы):

```
precmd () {
        [[ -t 1 ]] || return
        case $TERM in
                (sun-cmd) print -Pn "\e]l%~\e\\"
                ;;
                (*xterm*|rxvt|(dt|k|E)term) print -Pn
"\e]2;%~\a"
                ;;
        esac
}
```

Существует еще одна функция, **periodic()**, которая исполняется периодически, по истечении количества секунд, установленного в переменной **PERIOD**.

Есть удобная утилита *time*, которая показывает общее время CPU, использованное командой. Единственная проблема – не позабыть ввести **time** перед запуском команды. А что если вы запускаете весьма долгоиграющую команду и хотите определить продолжительность ее выполнения? В *Zsh* вы можете сделать это, задав в переменной **REPORTTIME** срок в секундах. Если команда превысит его, оболочка выведет информацию о времени выполнения по ее завершении. Формат выводимых сведений указывается в переменной **TIMEFMT**. Например:

```
REPORTTIME=5
TIMEFMT="%U user %S system %P cpu %*Es total"
```

Emacs среди оболочек

Вам должно быть уже ясно, что изучение *Zsh* – штука непростая. Зато *Zsh* и даст вам очень много, если не пожалеть времени на постижение основ. К счастью, есть множество отличных ресурсов, посвященных этой оболочке. Прежде всего это подробная система man-страниц. Кроме того, имеется онлайн-руководство Питера Стефенсона [Peter Stephenson] (доступное на http://zsh.sourceforge.net/Guide/zshguide.html). Оно немного устарело, но сохраняет свою ценность. Еще один ресурс для любителей *Zsh* – zshwiki.org. Итак, приступайте к настройке *Zsh* – вы очень выиграете, получив улучшенную оболочку. LXF

Темы приглашения

Zsh поставляется с набором тем оформления приглашения, очень простых в установке. Инициализируйте их:

```
autoload -U promptinit
promptinit
```

После этого вы сможете пользоваться командой **prompt** для выбора тем. Команда **prompt -p** обеспечит предпросмотр тем, а команда **prompt -s themename** назначит тему **themename** в качестве текущей. Если вам хочется выбирать тему случайным образом, скомандуйте **prompt -s random**.

```
Terminal
File  Edit  View  Terminal  Help
del% prompt -p

adam1 theme:
koan@del ~ % command arg1 arg2 ... argn

adam2 theme:
prompt_preview_theme:13: parse error
.-(~)-----------------------------------------------------------(koan@del)-
`--> command arg1 arg2 ... argn

bart theme:
del [prompt -p] ~                                        10-02-25  5:45PM
% command arg1 arg2 ... argn

bigfade theme:
   koan@del             Thu Feb 25 05:45:54pm
/home/koan> command arg1 arg2 ... argn

bigfade theme with parameters 'red white grey white':
   koan@del             Thu Feb 25 05:45:54pm
/home/koan> command arg1 arg2 ... argn

clint theme:
[Thu 10/02/25 17:45 CET][pts/0][i686/linux-gnu/2.6.31-19-generic][4.3.10]
<koan@del:~>
zsh 23 % command arg1 arg2 ... argn
```

› **Протестируйте темы командой prompt и упростите интерпретацию вывода с помощью встроенных функций *Zsh* по управлению цветами.**

Скорая помощь

Zsh имеет даже программу по проверке орфографии. Если вы активируете ее командой **setopt CORRECT** в вашем файле **zshrc**, оболочка будет обнаруживать ошибки (например, **chomd**) и просить вас их исправить.

Что за штука... Quickly

Пол Хадсон представляет быстрейший, простейший и непримиримейший метод для постижения этой штуки, называемой программированием.

Хм, опять о программировании. Мало, что ли, о нём говорилось?
Но ведь мы все любим программирование – многие именно из-за этого и пришли в мир Linux. На сей раз о Mono ни слова!

В самом деле?
Ну да! Quickly – ух ты! – построен вокруг Python, по крайней мере сейчас. Но это не просто отдельный язык программирования, инструментарий для пользовательского интерфейса или система контроля версий: проект Quickly задуман как мощный комплекс автоматизации практически любых процессов.

Нет уж, давайте о программировании, а не об этих, как вы их там назвали: я как-то не готов к разговору о них.
Не бойтесь: пока что Quickly – это именно средство программирования. У него немало сходства со стапельными [scaffolding] инновациями, представленными Ruby on Rails несколько лет назад.

Я сроду не пользовался Rails, и для меня это пустой звук. Что это за стапельные инновации?
Ну, пускай вам нужно написать простую программу для чтения RSS с графическим интерфейсом на *GTK*, причем на Python. Для этого придётся установить сам Python, библиотеки разработчика *GTK*, некоторые библиотеки XML (чаще всего *libxml2*), затем составить базовый код только для того, чтобы подготовить *GTK* к полезной работе. Возможно, понадобится установить и *Glade* (дизайнер пользовательского интерфейса *GTK*), затем собрать базовую оконную оболочку вашей будущей программы.

Ну и что здесь плохого?
Это не плохо, но скучно. И много повторных действий. А теперь Quickly их отменяет: после его установки командой **sudo apt-get install quickly** отовсюду будет автоматически стянуто около 40 МБ зависимостей, создающих базу для программирования. То есть, пользуясь Quickly при разработке приложений, вы получаете нечто вроде стапелей – строительных подпорок вашего приложения, включая подготовку *GTK* и, при желании, *Glade* для программирования интерфейса.

Да *Glade* настроить не так уж и трудно...
Речь не о тяжести труда, а о быстроте: Quickly сворачивает час нудной работы в единственную команду **quickly create ubuntu-project**.

Стоп, вы, кажется, упомянули Ubuntu? Я распрощался с ним после того, как там без моего ведома переставили все кнопки.
Quickly – это проект Canonical, что становится ясно вскоре после начала работы в нём. Вспомните о первой версии Ubuntu: там был один браузер, один офисный пакет, один текстовый редактор – и так далее. Разработчики сами решали, что будет лучше для пользователя, и для того времени это было революционным решением.

Это характерно для методологии Ubuntu. Несколько месяцев назад глава Canonical Марк Шаттлворт выразился так: «Мы намерены занять непримиримую позицию; будем применять основные принципы проектирования и сами отвечать за результат». Quickly следует тому же курсу по части программирования: ради ускорения процесса некоторые вещи выбирают за вас.

> «В отличие от Mono или Vala, Quickly не создает зависимостей.»

И что выбирают?
Как я уже говорил, Python. А также *GTK* и *Glade*. В качестве системы контроля версий предлагается *Bazaar*, а для обнародования своей программы можно воспользоваться Launchpad, специальным репозиторием Ubuntu.

Что-о?! Ненавижу *GTK*! И *Bazaar* тоже!
Имеете право. Это распространённое следствие непримиримых позиций – не каждый захочет взять Quickly на вооружение прямо сейчас. Но те, кто способен примириться с чужим выбором, существенно нарастят продуктивность.

То есть те, кому это не нравится, могут выйти вон?
Отнюдь нет. Предопределения касаются только шаблона **ubuntu-project**. А вы можете создать собственный шаблон, включить в него всё, что угодно, а ненужное зачеркнуть: *Glade*, *Bazzaar* или даже сам Python (хотя я и не пойму, за что...).

А если я применю Quickly, то буду навечно к нему прикован?
Опять нет. В отличие от Mono или Vala, Quickly не создает зависимости от кода или среды времени выполнения: это только средство сборки остова для готового проекта, само это средство можно перестраивать в широких пределах. Если вы однажды решите дать Quickly отставку, на здоровье – ни один проект от этой системы не зависит.

Предположим, я сживусь с Quickly... Похоже, он помогает только на начальном этапе. А будет ли польза от него на более поздних стадиях проекта?

Вы удивитесь: будет! Смирившись с *Bazaar*, вы с ходу получите мощную систему контроля версий (отслеживания истории ваших файлов), безо всяких усилий. Это развитый инструмент, который сохраняет и регистрирует каждый сделанный вами шаг. Если вы поймёте, что сделали ошибку, и захотите вернуться к прежней версии, *Bazaar* вам поможет, хотя не помешает сопровождать сделанные изменения комментариями типа «удвоил количество примочек в файле **foobar.py**».

Круто, конечно, но система контроля версий меня не впечатляет. Что ещё?

Как вам нравится создание пакета Debian одной командой? Правда, придётся ещё установить пакет **devscripts**, да и сама команда выполняется где-то с минуту. Но согласитесь: разве не здорово так просто передать свою программу другу? Достаточно отправить электронное сообщение с Deb-пакетом, который для вас соберёт Quickly!

Но я мизантроп — живу как сыч, и у меня нет друзей!

ОК. Ну, не совсем ОК – это всё-таки странно. Но вы все равно можете подарить свою работу людям: наберите простую команду **quickly release** для выгрузки вашего пакета на Launchpad в виде персонального архивного пакета (PPA), и назавтра проснётесь знаменитым.

А если меня не устраивает Launchpad?

И снова вопрос выбора. Не вашего выбора, а автора шаблона **ubuntu-project** в проекте Quickly. Если он вам не нравится – создайте свой шаблон, включив в него всё, что захотите.

Это сложно?

Индивидуально. Шаблоны в Quickly – просто рабочие скрипты, их можно писать на Python, Perl или на чём хотите. Простейший метод сборки собственного шаблона – загрузить готовый (лучше всего **ubuntu-project**, как наиболее универсальный) и изменить в нём участки, которые вас не устраивают. Скомандуйте **quickly quickly ubuntu-template my-awesome-template** – и меняйте себе. Хотите по умолчанию Mono или C# – флаг вам в руки!

Вы вроде сказали, что новый проект сгодится не только для программирования?

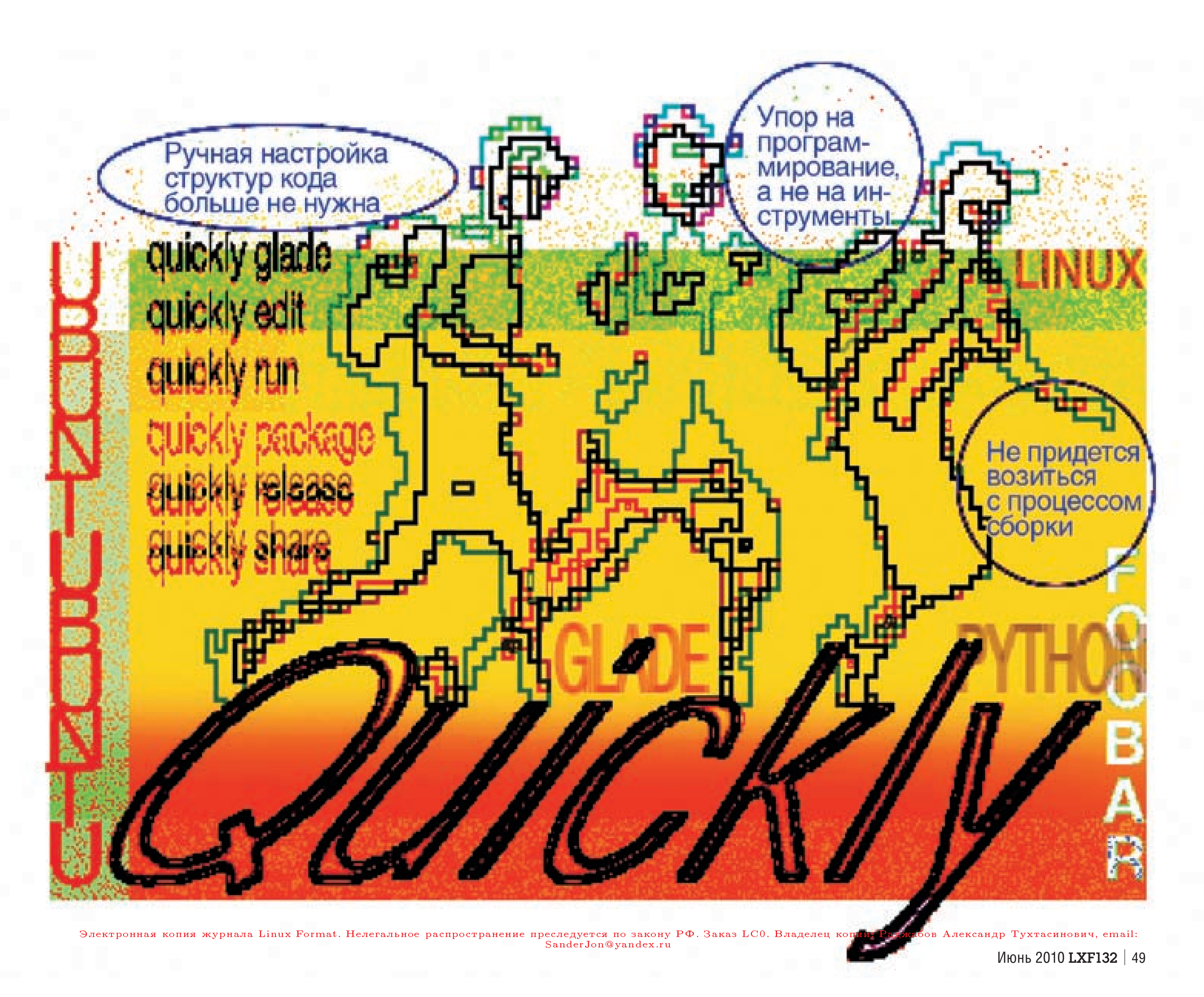
Да, я не вижу причин, по которым нельзя скроить шаблон для чего-нибудь иного. Например, если вы постоянно пользуетесь одним и тем же набором файлов для издания ежемесячного бюллетеня, Quickly послужит каркасом для сборки каждого нового номера.

Хм, ладно. Попробую. Не обещаю, что мне понравится, но попробовать можно. Не дадите ли пару ссылок для начала?

Установив Quickly, наберите команду **quickly tutorial ubuntu-project** – загрузится краткое вводное руководство. Существует видеоролик, разработанный авторами Quickly, который наглядно описывает процесс создания (несколько рудиментарной) программы: www.youtube.com/watch?v=dx93U10Kkro.

Рудиментарной? А подробного руководства, с пошаговым прохождением всего процесса, разве нету?

Пока нет, но мы подготовим его специально для вас в одном из следующих номеров – не пропустите! LXF





По рецептам доктора Брауна

Д-р Крис Браун
Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Добротное администрирование систем из причудливых заворотов кишок серверной.

Верните das blinkenlights

Первые компьютеры, на которых я работал, были намного милее современных. На них были ряды лампочек, и при большом желании можно было пошагово выполнять программу и видеть содержимое каждого регистра, цикл за циклом. Это была эпоха das blinkenlights – проблесковых индикаторов, и почти в каждом компьютерном зале лежала выцветшая фотокопия надписи крупным готическим шрифтом:

«Дас машине не есть для тыкирен пальцен кому попаль. В противный слютшай она легко быть пофрешден. Дас машине ист фюр профессионален работен унд нихт фюр придуркен. Кто желать смотрирен, держаль школдивый ручонка в карман. Расслабирен и глазель на blinkenlights».

Лампочки были даже на ранних любительских компьютерах типа Altair 8800 (1975 г.), но после Commodore PET (1977 г.) исчезли навсегда.

Исчезающий свет

Сегодня принцип «лучше меньше, да лучше» стал мантрой для мигающих лампочек. Мой новый сервер так стыдится светодиодов на передней панели, что их совершенно невозможно найти, когда он выключен. На моем домашнем хабе четыре прелестных трехцветных светодиода, но увы – они не мигают. И когда хаб работает нормально (а это, к счастью, 99,9 % всего времени), никакой полезной информации они не несут.

Светодиоды на моем ноутбуке так утонченно уменьшены в размерах, а их свет так близок к синему концу видимого спектра, что приходится ждать сумерек, чтобы понять, включен ли Caps Lock. Чтобы понаблюдать за активностью сети, я должен приложить левое ухо к столу и скосить глаза на крошечный светодиод, утопленный в RJ45. И во всем моем офисе нигде нет огонька, который бы показывал активность беспроводной сети.

Так что каспада, пошалюйста, ферните немношко мигающих лампочек! Пожалуйста!

Сервер Lucid Lynx

Ubuntu 10.04 Новый релиз Ubuntu с долгосрочной поддержкой – гигантский прыжок для человечества?

Большинство из нас считают Ubuntu дружелюбным настольным дистрибутивом, но в Canonical есть команда, усиленно работающая над тем, чтобы он мог составить конкуренцию Red Hat на серверном рынке. Ubuntu 10.04 LTS Lucid Lynx появится на диске следующего номера **LXF**, и это один из релизов «с долгосрочной поддержкой», к которым Canonical относится очень серьезно. Для серверного продукта «долгосрочный» означает поставку обновлений и поддержку в течение пяти лет. Учебные курсы для этого дистрибутива построены на релизах LTS, и скоро должны появиться новые версии Certified Professional, Server и курсов по облачным вычислениям. До Lucid Lynx можно обновиться с предыдущего «обычного» релиза (9.10) и с последнего релиза LTS (8.04). Итак, Ubuntu 10.04 – важная веха, и как серверный продукт он далеко ушел от двухлетнего Hardy Heron. Вот часть новинок дистрибутива.

Облачные вычисления в 10.04 повсюду. В первом установочном меню появляется подсказка о возможности установить узел Ubuntu Enterprise Cloud (UEC). Экраны разбиения диска на разделы также отличны от 8.04: по умолчанию создается файловая система ext4 поверх LVM. Можно зашифровать каталог **home** и/или соответствующий ему раздел. Есть и новые профили установки, которые облегчают развертывание серверов, выполняющих различные роли в облаке Eucaliptus.

Менее явные изменения – ядро стало поддерживать рандомизацию адресного пространства (случайным образом меняется расположение основных областей данных внутри процесса, сильно затрудняя определение целевых адресов злоумышленником). Ядро 2.6.32 поддерживает технологию агрегации памяти Kernel SamePage Merging (KSM), позволяющую нескольким процессам с одинаковыми страницами памяти совместно использовать одну общую страницу. Это особенно удобно, когда в *KVM* работают несколько виртуальных машин с одинаковыми ядрами.

› **Canonical поддерживает обновления между «обычными» релизами Ubuntu и версиями с «долгосрочной поддержкой».**

Другие новинки

Множество пакетов, обновление или появление которых ожидалось с 8.04 LTS, включает *Django*, *CouchDB*, *RabbitMQ* (система обмена сообщениями на основе AMQP), Eucaliptus, *etckeeper* (система контроля версий для **/etc**, которая внедряется в утилиты *APT*), *Tomcat 6*, *OpenJDK 6*, *MySQL 5.1*, *ClamAV*, *SpamAssassin*, *Puppet*, *AutoFS5*, *Samba* 3.4, Ruby, Python 2.6, Erlang, *Nagios 3* и *Corosync* («кластер» с открытым исходным кодом).

Частный детектив в Интернете

IP-сыщик Д-р Браун идет по следам киберпреступника.

Загляните в файлы журналов любого компьютера, открытого для доступа извне, и вы найдете множество неудачных попыток входа в систему, вызванных автоматизированными атаками скриптов. Если порт 22 открыт, большинство атак будут попытками входа через SSH. Ежедневно просматривая журналы безопасности web-сервера, я видел преимущественно два типа неудачных входов в систему:

```
Mar 22 15:09:53 ns sshd[9446]: Failed password for invalid user ns from 174.121 .23.114 port 50963 ssh2
```

Здесь атакующий ищет подлинные имена пользователей. В тот день журнал зафиксировал 13647 таких атак с 959 различными именами пользователей. Второй тип выглядел так:

```
Mar 22 08:40:45 ns sshd[23270]: Failed password for root from 222.122.163.82 port 56501 ssh2
```

Это неудачная попытка входа в систему от имени суперпользователя-root, и таких попыток за 10 минут было 258. Заметьте: оба сообщения содержат IP-адрес компьютера, с которого проводились атаки. Собственно говоря, 57 % из 99647 строк журнала содержали IP-адрес **222.122.163.82**. Посмотрим, что мы сможем о нем узнать. Пока вы радостно предвкушаете, как мы раскроем имя и адрес плохого парня, немного охладим ваш пыл. Во-первых, найденный IP-адрес может быть адресом шлюза NAT, во-вторых, многие атаки проводятся со взломанных компьютеров.

За кулисами IP-адреса

Попробуем сделать по IP-адресу обратный DNS-запрос. Результат выполнения команды немного сокращен.

```
$ dig -x 222.122.63.82
;; QUESTION SECTION:
;82.163.122.222.in-addr.arpa. IN
PTR
;; AUTHORITY SECTION:
163.122.222.in-addr.arpa. 600 IN
SOA
ns1.ne.kr.admin.100dedi.com.
```

PTR-записи не найдено, но мы узнали, что сервер имен для этого блока адресов находится в Корее. Можно также заглянуть в базу данных RIPE на www.db.ripe.net с помощью **whois**:

```
$ whois 222.122.163.82
KRNIC is not an ISP but a National Internet Registry similar to APNIC.
The following is organization information that is using the IPv4 address.
IPv4 Address          : 222.122.163.0-222 .122.163.255
Network Name: KORNET-INFRA000001
Connect ISP Name      : KORNET
Registration Date     : 20100126
Publishes : N
[ Organization Information ]
Organization ID       : ORG1600
Org Name : Korea Telecom
Address   : Jungja-dong, Bundang-gu, Sungnam-ci
Zip Code   : 463-711
```

Теперь мы знаем, что провайдер атакующего – Korea Telecom. Программа *geoiplookup* сообщит его географическое положение:

```
$ geoiplookup 222.122.163.82
GeoIP Country Edition: KR, Korea, Republic of
```

За обновления базы данных GeoIP нужно платить; платить также можно за запросы через Интернет – см. www.maxmind.com. Полезные подробности иногда можно получить с помощью *traceroute*. Посмотрите на временн*ы*е интервалы прохождения запроса, сравните их с временн*ы*ми интервалами прохождения запроса туда-обратно, предоставленными командой *ping*, и, может быть, вам удастся определить положение маршрутизатора рядом с атакующим компьютером.

Несколько полезных утилит для розысков в Интернете есть на сайте www.robtex.com. Можно узнать, какие сайты находятся на заданном IP-адресе, кто владеет блоком IP-адресов, добавлен ли сайт в черный список и многое другое. (Следует подчеркнуть, что нет причин подозревать владельцев сайтов, размещенных на этом компьютере. Они почти наверняка лишь невинные свидетели).

Наконец, можно попробовать просканировать компьютер командой *nmap*:

```
# nmap -O 222.122.163.82
Interesting ports on 222.122.163.82:
Not shown: 985 closed ports
PORT STATE SERVICE
21/tcp open ftp
22/tcp open ssh
53/tcp open domain
80/tcp open http
.. . вывод опущен ...
OS details: Linux 2.6.9 - 2.6.15, Linux 2.6.9 - 2.6.15 (x86),
Linux 2.6.9 - 2.6.20
```

Открытые порты говорят нам, что это типичный web-сервер с Linux. Итак… преступник, вероятно, живет в Сеуле, пользуется Linux, прихрамывает и курит «Кэмел». Элементарно, Ватсон!

Задраиваем люки

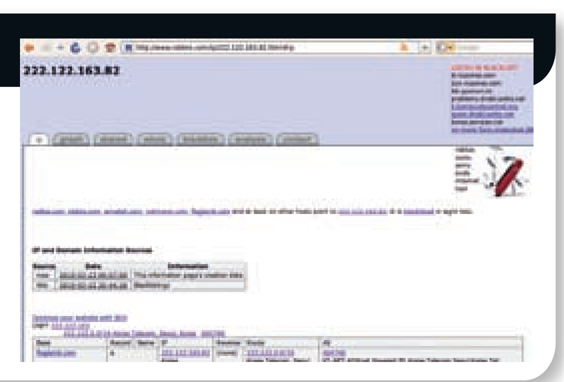
Возможный ущерб от атак на SSH можно снизить, запретив прямой вход в систему от имени root – просто убедитесь, что в **sshd_config** есть следующая строка:

```
PermitRootLogin no
```

Другая технология, особенно эффективная против атак прямым перебором – анализ сообщений, которые активно пишет в журнал сервер SSHD, и запрет доступа с IP-адресов, откуда производятся неудачные попытки входа в систему.

Это делает скрипт *DenyHosts* – он блокирует адреса, помещая их в **/etc/hosts.deny**.

Fail2ban также запускается по анализу записей из журналов, но действует через обновление правил брандмауэра в *Iptables*.



› На сайте Robtex можно узнать провайдера атакующего компьютера и сайты, которые на нем находятся. Внимание: черный список!

Настраиваем ProFTPD

FTP-серверы Установим профессиональный FTP-сервер и настроим анонимный или аутентифицированный доступ с полнофункциональным *ProFTPD*

На прошлой неделе меня поймал врасплох один из клиентов, просивший помочь в установке FTP-сервера. Я этим давно не занимался, и некогда тесное знакомство с FTP почти улетучилось из памяти. Подозреваю, что дни FTP сочтены, и HTTP более созвучен современности.

Однако недостатка FTP-серверов в обычных репозиториях нет: в Fedora их два (*ProFTPD* и *vsftpd*), а в Ubuntu – не менее четырех. Дальнейший поиск с помощью монеты и «звонка другу» помог остановить выбор на *ProFTPD*, и на данном уроке я опишу свой опыт по его установке и настройке. Я делал это в *Fedora 12*, но в других дистрибутивах все должно быть примерно так же.

ProFTPD – мощный полнофункциональный FTP-сервер. Он запускается как самостоятельная служба или через *xinetd* и поддерживает анонимный и аутентифицированный вход в систему, включая «виртуальных пользователей», учетные записи которых определены в каталогах LDAP или базах данных SQL. Он также допускает виртуальный хостинг – обслуживание нескольких FTP-сайтов с одного экземпляра FTP-сервера. Рассмотрим все эти возможности.

Сначала я установил сервер и настроил его запуск во время загрузки:

```
# yum install proftpd
# chkconfig proftpd on
```

Сразу после установки сервера инстинкт погнал меня к файлу настройки (в данном случае **/etc/proftpd.conf**). Его структура с контейнерными XML-тэгами подобна файлу настройки *Apache*. В файле настройки по умолчанию, который поставляется с *Fedora 12*, находится рабочая конфигурация с рядом добавочных опций, которые закомментированы. На **рис. 2** показан минимальный файл настройки. Он содран с поставляемого (урезан), но достаточен для поддержки аутентифицируемого входа по FTP.

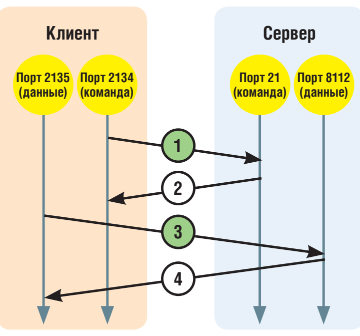
Анонимный FTP

Анонимный FTP позволяет пользователю аутентифицироваться под именем «ftp» или «anonymous» и паролем, которым по традиции должен быть его e-mail адрес, но на практике это не проверяется. Анонимный FTP широко применяется для предоставления доступа к интернет-ресурсу только на чтение и на много лет предвосхитил использование для тех же целей HTTP и web-сайтов. Анонимный вход в систему снимает мои заботы по поводу безопасности FTP. Во-первых, по сети не передаются настоящие имя и пароль, которые могут перехватить плохие парни; во-вторых, факт отсутствия шифрования соединения не имеет значения, ведь содержимое сервера все равно публично!

Клиент
Сервер
Порт 2135 (данные)
Порт 2134 (данные)
Порт 21 (команда)
Порт 20 (данные)
1
2
3
4

Активный режим FTP

› **Рис. 1. В активном режиме FTP-сервер инициирует канал данных с клиентом (шаг 3), используя номер порта, переданный ему клиентом на шаге 1. Этот канал скорее всего будет блокирован шлюзом клиента.**



Пассивный режим FTP

› **В пассивном режиме канал данных инициирует клиент (шаг 3), используя номер порта, переданный ему сервером на шаге 2. Этот вариант «дружелюбнее» по отношению к брандмауэру.**

В файле **proftpd.conf**, который по умолчанию поставляется с Fedora 12, также есть секция **<Anonymous>**, которая работает сразу же. Ну, почти. Мне понадобилось только добавить в файл **/etc/sysconfig/proftpd** следующую строку:

```
PROFTPD_OPTIONS="-DANONYMOUS_FTP"
```

и перезапустить *ProFTPD* командой:

```
service proftpd restart
```

В конфигурации по умолчанию каталог **~ftp** (которому соответствует **/var/ftp**) является каталогом **home** для аноним-

Некролог для FTP?

FTP – протокол древний. Первые RFC датируются 1971 годом, а формально он был определен в RFC454 (1973) и RFC959 (1985). Приблизительный современник Telnet, FTP был изобретен задолго до брандмауэров, частных адресных пространств, NAT и прочих сомнительных удовольствий современного Интернета, и не очень хорошо с ними ладит. Начнем с того, что FTP использует два различных соединения – для данных и для управления. Он работает в «активном» или «пассивном» режимах. Эти термины описывают ситуацию с точки зрения сервера, отражая его роль: инициирует ли он канал данных с клиентом или пассивно принимает его от клиента (см. **рис. 1**). В активном режиме серверу почти наверняка не удастся сделать это из-за фильтрации и маскирования пакетов, выполняемых шлюзом, через который клиент подключен к Интернету. Пассивный режим проще, но ему все равно необходим отдельный порт (на сервере) для каждого клиента, и потенциально на брандмауэре сервера должно быть открыто множество портов.

В FTP нельзя задать полное имя (FQDN) сайта, с которым производится соединение, что делает невозможным виртуальный хостинг на основе имен. Механизм аутентификации, при котором имя и пароль (да и сами файлы) передаются в открытом виде, в нынешнем озабоченным безопасностью Интернете безнадежно устарел.

Часть этих трудностей можно обойти перенаправлением локального порта SSH для создания защищенного туннеля для FTP-траффика – и есть кое-какие удобные хитрости при отслеживании соединения, которые помогут брандмауэрам приспособиться к необычному использованию портов в FTP. Но все это напоминает мне Главный дуб в Шервудском лесу, давно уж навалившийся на ограду. При виде него я порываюсь из жалости выдернуть подпорки и дать почтенному дереву упасть и достойно умереть, чтобы по законам природы на его месте появился новый росток.

ных пользователей. При установке пакета ProFTPD он создает каталоги **/var/ftp/pub** (куда нужно выкладывать контент) и **/var/ftp/upload** (куда могут выкладывать контент анонимные пользователи).

Конфигурация по умолчанию также поддерживает аутентификацию для пользователей, у которых на сервере есть обычные учетные записи. После входа по FTP вы попадаете в свой домашний каталог. Сначала мне это не удалось, и я некоторое время искал причину, запустив *ProFTPD* не в режиме демона, со включенной отладкой:

```
proftpd -n -d 5
```

и параллельно запустив **tail -f** на **/var/log/messages**. В итоге оказалось, что SELinux не давал демону зайти в домашний каталог, и я поступил, как всегда в таких ситуациях: перевел SELinux в разрешающий режим (permissive mode):

```
setenforce permissive
```

после чего все заработало. На настоящем сервере относиться к уровню безопасности SELinux нужно более серьезно.

Виртуальный пользователь – пользователь, у которого нет обычной учетной записи Linux на сервере в **/etc/passwd** или где-либо еще. Понятие «виртуальный пользователь» вводит в заблуждение – некоторые из них вполне реальны: настоящее тепло, дышащие тела; виртуальны только их учетные записи. Эти записи могут храниться в **AuthUserFiles**, каталогах LDAP и базах данных SQL. Простейший механизм аутентификации виртуального пользователя – **AuthUserFile**, это простой текстовый файл в том же формате, что и **/etc/passwd**.

Виртуальный хостинг

А если нужно разместить несколько FTP-сайтов в одной системе? К сожалению, реализовать такой же виртуальный хостинг, как и для web-сайтов, когда записи нескольких узлов в DNS ссылаются на один и тот же IP-адрес, здесь не удастся. Это работает для HTTP, но не для FTP, потому что в HTTP-запросе есть поле «Host», поясняющее web-серверу, для какого сайта предназначен запрос, даже если все они приходят на один и тот же IP-адрес. В протоколе FTP такого поля нет, и в виртуальном хостинге FTP у каждого сайта должен быть собственный IP-адрес, по которому сервер мог бы понять, для какого сайта предназначен запрос.

> «Относиться к уровню безопасности SELinux нужно серьезно.»

Но даже если вы счастливый обладатель достаточного количества свободных IP-адресов, нужно сделать еще две вещи. Во-первых, определить «IP-алиас» на сетевой карте для всех адресов, которые будут использоваться. Во-вторых, создать для каждого сайта в файле **proftpd.conf** секцию **<VirtualHost>**.

В дистрибутивах на базе RedHat для каждого дополнительного IP-адреса, который нужно связать с сетевой картой, понадобится создать файл с именем вроде **/etc/sysconfig/network-scripts/ifcfg-eth0:N**, где N – номер алиаса (0, 1, 2, ...). Файл будет примерно таким:

```
DEVICE=eth0:1
BOOTPROTO=static
IPADDR=192.168.1.69
ONBOOT=yes
HWADDR=00:0C:29:E1:E4:23
```

Замените значения **DEVICE**, **IPADDR** и **HWADDR** соответствующими в вашей системе. Для каждого IP-адреса понадобится отдельный файл. Насколько я знаю, сделать нечто вроде «Я хочу назначить все адреса из блока 10.1.0.0/24 этой сетевой карте» нельзя (если я неправ, сигнализируйте!).

После создания файлов выполните команду

```
service network restart
```

```
# Server Config - config used for anything outside a <VirtualHost> or <Global> context

ServerName                  "ProFTPD server"
ServerIdent                 on "FTP Server ready."
ServerAdmin                 root@localhost
DefaultServer               on

# Cause every FTP user except adm to be chrooted into their home directory
VRootEngine                 on
DefaultRoot                 ~ !adm
VRootAlias                  etc/security/pam_env.conf /etc/security/pam_env.conf

# Use pam to authenticate (default) and be authoritative
AuthPAMConfig               proftpd
AuthOrder                   mod_auth_pam.c* mod_auth_unix.c

# Don't do reverse DNS lookups (hangs on DNS problems)
UseReverseDNS               off

# Set the user and group that the server runs as
User                        nobody
Group                       nobody

# Define the log formats
LogFormat                   default "%h %l %u %t \"%r\" %s %b"
LogFormat                   auth    "%v [%P] %h %t \"%r\" %s"

# Global Config - config common to Server Config and all virtual hosts
<Global>
  Umask                     022

  # Don't do ident queries (hangs when the port is filtered)
  IdentLookups              off

  # Allow users to overwrite files and change permissions
  AllowOverwrite            yes
  <Limit ALL SITE_CHMOD>
    AllowAll
  </Limit>

</Global>
```

› Рис. 2. Этой минимальной конфигурации достаточно для поддержки аутентифицированного входа по FTP.

чтобы подцепить новые адреса, и проверьте, что сетевые интерфейсы для всех адресов созданы, командой

```
ifconfig
```

Для каждого из сайтов в файле настройки необходима секция **<VirtualHost>**. Она должна выглядеть примерно так:

```
<VirtualHost 144.16.44.5>
...
</VirtualHost>
```

или вместо IP-адреса можно использовать имя компьютера

```
<VirtualHost foo.example.com>
...
</VirtualHost>
```

но это не должно ввести вас в заблуждение. Имя компьютера будет разрешено в IP-адрес, а он-то и определяет, какой FTP-сайт будет обслуживаться. Если у вас несколько имен компьютеров, которые DNS преобразует в один и тот же IP-адрес, и у каждого из них есть свой раздел **<VirtualHost>**, будет выбран первый из них. Если у нескольких хостов есть что-то общее, можно вынести это в раздел **<Global>** файла настройки.

Чтобы это доказать, я назначил второй IP-адрес (192.168.1.69) сетевой карте FTP-сервера и добавил раздел в **proftpd.conf** – вы можете найти ее на нашем DVD. «Второй» FTP-сервер предоставляет анонимный доступ к каталогу **/var/ftp/secondary/pub**. Я также создал файл **/var/ftp/secondary/welcome.msg** с приветствием.

Многие возможности *ProFTPD* остались за рамками нашего обзора – они включают управление доступом, ведение журналов (см. стр. 32) и «песочницы» *chroot*. LXF

Где узнать больше

Сайт *ProFTPD* находится на www.proftpd.org.

Список директив файла настройки, по алфавиту, см. на сайте www.proftpd.org/docs/directives/linked/by-name.html.

Интересное обсуждение активного и пассивного режимов FTP – на http://slacksite.com/other/ftp.html.

Чтобы узнать больше о виртуальных пользователях, можете заглянуть в руководство по созданию учетных записей с помощью *MySQL*, www.howtoforge.com/proftpd_mysql_virtual_hosting. Также взгляните на www.kreci.net/linux/proftpd-virtual-user-quick-howto: там приводится сурово-краткое объяснение **AuthUserFile**.

Организатор:
Академия Информационных Систем



ВАЖНОЕ ДЕЛОВОЕ СОБЫТИЕ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ СОЧИ-2010

Дата проведения: с 7 по 11 сентября 2010 г.
Место проведения: г. Сочи, Отель "Рэдиссон САС Лазурная"



9 ЕЖЕГОДНАЯ ВСЕРОССИЙСКАЯ КОНФЕРЕНЦИЯ

«ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ. РЕГИОНАЛЬНЫЕ АСПЕКТЫ»



За 8 лет конференция стала де-факто ежегодным съездом профессионалов по информационной безопасности России. За прошедшие годы свыше 1500 делегатов приняло участие в конференции, среди которых руководители и специалисты государственных и коммерческих организаций, научных учреждений и общественных объединений.

В конференции традиционно принимают участие представители федеральных органов государственной власти, законодательных, контролирующих и регулирующих структур.

На конференции обсуждаются вопросы государственного регулирования, регионального развития, отраслевые проблемы, развитие систем управления и технологий в области защиты информации.

Регистрация участников проводится до 1 сентября 2010 г.





По вопросам участия обращайтесь:
Академия Информационных Систем,
г. Москва, ул. Первомайская, д. 126
Тел./факс: (495) 231-38-74
E-mail: conf@infosystem.ru
www.vipforum.ru
www.infosystem.ru













LINUX FORMAT Учебники

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

Свобода выбирать отнюдь не бесплатна

Хотите подарить людям свободу?
А что они с ней будут делать?
Из фильма по мотивам пьесы
Евгения Шварца «Убить дракона»

Компания Wolfire опубликовала статистику по объёму выплачиваемых пользователями различных операционных систем сумм за набор из пяти игр от независимых разработчиков, который раздавался по принципу «плати, сколько хочешь». Представителей от «царства Linux» было около 14 %, но доход от них составил около 23 % от общей суммы. Иными словами, пользователей Linux пока ещё мало, но они готовы платить – в том числе и за дополнительное удовольствие. Свобода того стоит, и современные адепты свободного окружения это вполне осознают.

Да, когда нас будет много, мы ничем не будем отличаться от среднего пользователя компьютера, но и сейчас наше число не такое уж и маленькое, а в дополнение к этому мы почти в два раза сознательнее, чем «средняя температура по компьютерной больнице». Издатели и создатели, учтите этот факт на будущее, когда будете решать вопрос о выпуске программного обеспечения под GNU/Linux.

P.S. Вышел *Emacs 23.2*. Он стал ещё лучше, и текстовый редактор, как бы ни фыркали завистники, в нём близок к совершенству, как никакой другой.

P.P.S. Hewlett-Packard подхватила остатки Palm и будет вкладываться в webOS, базирующуюся на Linux. Надеюсь, «допилят», так как Google с его *Android* необходим качественный конкурент.

E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...

Совет месяца: Калькулятор по требованию



Диалог «Выполнить команду» [Run Command], вызываемый по Alt+F2, в KDE 4 значительно преобразился. Он по-прежнему умеет, хм, выполнять команды, открывать всякие URL и искать по web-сайтам (соответствующие «волшебные слова» приведены в настройках *Konqueror*), но его возможности обогатились.

Поиск по рабочему столу – вещь достаточно известная, и по сути она означает, что вы можете искать не только по именам, но и по содержимому файлов, метаданным документов, контакт-листам и даже заголовкам открытых окон. Вы также можете найти музыкальные композиции и тут же добавить их к своему плей-листу. Если вам и этого мало, *KRunner* (так теперь называется данный диалог) способен выступать в роли калькулятора: наберите в нем «2+2=» или «=2+2», и увидите результат.

Данный функционал не ограничен простейшей арифметикой – можно использовать тригонометрические и прочие функции. *KRunner* также умеет преобразовывать единицы измерения: наберите **5 in>cm**, чтобы узнать, чему равны 5 дюймов в сантиметрах (обратите внимание на пробел между **5** и **in**). Как и следует ожидать, поддерживается широкий набор единиц измерения: мы не нашли фурлонгов или кабельтовых, но световые года и прочие парсеки присутствуют.

Этот маленький диалог способен на очень многое; нажмите на кнопку ? в окне, чтобы получить список всех возможностей, которые он предоставляет.

Inkscape: Простой

Нужна графика для презентации, рассылки или сайта? **Энди Ченнел** научит быть креативным с помощью *Inkscape* и библиотеки *OpenClipArt Library*.





Наш эксперт

Энди Ченнел
Энди, видимо, будет делать Первые шаги в Linux вечно, а технологиями он заинтересовался, открыв для себя Dragon 32.

Когда вы представляете какую-либо работу, идею или проект, вид его зачастую не менее важен, чем сам материал. На данном уроке мы собираемся использовать для создания типографики и оформления открытый графический пакет *Inkscape*. Мы создадим несколько простых иконок, воспользовавшись иллюстрациями библиотеки *OpenClipArt*.

OpenClipArt 2.0 доступна по адресу www.openclipart.org. Изображения представлены в пакетах от 350 МБ до 420 МБ (в зависимости от используемого сжатия), и после загрузки и распаковки все они представляются в виде набора каталогов-категорий, вместе с HTML-страницами, чтобы вы могли просмотреть, как они выглядят. Библиотека доступна в виде пакета формата PNG/SVG



› Здесь показаны два основных сгруппированных элемента в отдельности — эффект стекла слева и иконка справа.



› Многие элементы OpenClipArt были созданы в *Inkscape* как группа объектов. Разгруппируйте их, чтобы получить отдельные части.

или только SVG, и после распаковки занимает на жестком диске 1,2 ГБ. Мы выбрали комбинированную библиотеку, потому что наличие двух форматов дает больше возможностей: PNG служит для быстрого применения в проектах, а SVG-файлы подразумевают возможность редактирования и подгонки изображений для ваших целей.

Описание установки библиотеки *OpenClipArt* мы здесь приводить не будем; но вам следует позаботиться о наличии в вашей системе PHP. Загрузите и разверните архив **openclipart-2.0-tools** с http://download.openclipart.org/downloads/2.0 в каталог с распакованными изображениями и скомандуйте **make install** для сборки обозревателя. Наша версия *Inkscape* – 0.47, последняя включенная в Ubuntu, но любые другие недавние релизы также подойдут, поскольку мы не собираемся применять ничего такого эзотерического.

Редактируем изображение

Для начала импортируем в *Inkscape* изображение. Мы выбрали в библиотеке файл под именем **coredump_Glassy_WiFi_Signal.svg**. Это блестящая иконка; но почему бы не перекрасить ее из зеленой в пурпурную?

При щелчке на объекте в *Inkscape* он выбирается, и вокруг него появляются маркеры-рукоятки, расположенные в углах и по сторонам. Они служат для изменения размера объекта по горизонтали, по вертикали или в обоих направлениях одновременно. Если вы изменяете размер, но желаете сохранить пропорции оригинального изображения, то при перемещении маркеров удерживайте клавишу Ctrl. Щелкните на объекте вновь, и маркеры перейдут в режим «вращение и скручивание», позволяющий выполнять различные манипуляции с объектом.

Мы работаем с векторным изображением (оно описывается математически, а не составляется из пикселей), а значит, масштабирование, скручивание и вращение не приведут к потере качества – играйте с ними вволю. В *Inkscape* множество уровней отмены (Ctrl+Z), так что не бойтесь все запутать.

» **Месяц назад** Родительский надзор через *Glubble* и настройка рабочего стола.

дизайн графики

Если выбрать символ Wi-Fi и нажать иконку пурпурного цвета у основания окна, то все изображение станет пурпурным, поскольку на самом деле это группа отдельных элементов. Их можно разделить, щелкнув на изображении правой кнопкой мыши и выбрав Разгруппировать. В нашем случае у вас получится две группы. Первая содержит все тени и блики (стекло) – расположена сверху – а вторая содержит саму иконку. Таким образом, сам объект можно легко преобразовать в другую иконку, сохранив блеск.

Давайте оттащим стеклянную группу в сторону. Если при этом придержать клавишу Ctrl, группа переместится строго горизонтально; это упрощает последующее совмещение. Далее, разобьем иконку на части. Щелкните по ней правой кнопкой и выберите Разгруппировать (отметим, что, возможно, это придется делать дважды). Теперь выбирайте фон иконки. Щелкните на новом цвете в палитре и перенесите стеклянную группу обратно, поверх иконки.

Представляем слои

Многие проекты (например, кнопки для сайта) требуют набора согласованных изображений, редактируемых раздельно. В этом случае удобно править часть изображения, не мучаясь с растаскиванием элементов по сторонам, как делали мы. Слои – идеальное решение.

Оставаясь на изображении Wi-Fi, начнем с выбора стеклянной группы и нажмем Ctrl+X, чтобы вырезать ее. Теперь выберем Слой > Слои, чтобы увидеть палитру слоев у правого края экрана. Там будет одна запись с именем Layer 1. Щелкните на ней правой кнопкой мыши и выберите Переименовать слой, а затем введите **Icon**.

Далее воспользуемся кнопкой + для создания нового слоя и дадим ему имя Glass. Убедитесь, что он выбран, и нажмите Ctrl+V, чтобы вставить группу теней/бликов. Теперь можно перемещать стеклянную группу поверх иконки. Прелесть в том, что мы можем удалить эффект стекла, нажав на маленькой иконке глаза рядом с именем группы, и мы сможем поработать над иконкой, прежде чем включим Glass вновь для проверки эффекта.

Рядом с иконкой глаза располагается замок, предотвращающий перемещение и изменение слоя. Отлично будет заблокировать слой Glass и, не отключая его, редактировать слой Icon; тогда вы сразу же увидите эффект от изменений.



› **Размещая элементы в отдельных слоях, можно редактировать одну часть, не влияя на все остальное.**



› **Инструмент Текст в *Inkscape* работает так же, как в большинстве других приложений.**

Слоев можно создать сколько угодно и при необходимости перемещать их вверх и вниз по стопке при помощи кнопок-стрелок внизу палитры слоев. В нашем примере, мы отделим фон от самой иконки, чтобы иметь возможность сохранить и его, и слой Glass, а изменять только саму иконку – скажем, добавим на нее текст.

Отключите слой Icon, щелкнув на иконке глаза, и используйте кнопку + для создания нового слоя с именем Text. Его необходимо упорядочить, разместив между фоном и Glass. Для добавления текста примените инструмент Текст, обозначенный на панели инструментов слева буквой T. Щелкните где-нибудь на странице и начните набор. После ввода текста выделите его мышью и обратитесь к панели инструментов Текст (которая появится над рабочей областью) для изменения шрифта, начертания и размера. Можно изменить еще и цвет текста, выбрав оттенок на палитре внизу окна.

Одна из уловок, использующаяся для «поднятия» текста над кнопкой – добавление тени, а в *Inkscape* это делается весьма просто. Выберите текст инструментом Выделение – стрелочкой – а затем выполните Фильтры > Свет и тень > Отбрасываемая тень. В диалоге вы увидите опции для определения смещения, размывания и прозрачности. При их изменении используйте кнопку Предпросмотр, чтобы видеть эффект ваших изменений. Когда все будет готово, нажмите кнопку Применить. Отметим, что диалог не закроется, но эффект будет применен. Если вы щелкнете вновь, эффект станет уже двойным.



Поскольку на экране эти три кнопки должны быть довольно маленькими, выберите цвет шрифта, хорошо контрастирующий с фоном.

Добавим пару слов

Теперь мы готовы к завершению. Если вы делаете это для сайта или презентации *Impress*, то, вероятно, захотите выбрать Файл > Экспортировать в растр. Это приведет к запуску весьма сложного диалога. Обычно в верхней части просто указывается «Рисунок», а затем используется кнопка Выбрать вблизи нижнего края окна для определения места сохранения результирующего PNG-файла. После всех настроек нажмите кнопку Экспорт для получения финального результата.

Если вы делаете объект для другого проекта *Inkscape*, можете сохранить его в виде SVG-файла при помощи Файл > Сохранить как, и перейдя затем в соответствующий каталог. Изменяя слово слоя Text (как показано выше), можно получить множество различных версий одного и того же файла, готового к вставке в вашу презентацию или сайт. Вы даже сможете изменить цвет фона для создания сменяющихся состояний в более сложных проектах.

Impress: Даешь

Программа для презентаций в *OpenOffice.org* не только отображает слайды…

Кроме текстового процессора и электронной таблицы, в *OpenOffice.org* есть пакет подготовки презентаций *Impress*. Он превосходно справляется с созданием слайд-шоу, но способен и на действия поинтереснее. На данном уроке мы создадим небольшое самодостаточное *Impress*-приложение с графикой и ссылками, пригодное для использования почти во всех компьютерных системах. Наш проект будет представлять собой набор всплывающих карточек с фразой на французском, иллюстрацией и двумя английскими переводами. Затем мы попросим пользователя выбрать соответствующий перевод, и, в зависимости от ответа, он увидит, прав он или нет. Окна с результатом приведут либо к следующему вопросу, либо к завершению. Используемое здесь изображение Эйфелевой башни создал Стивен ван Вель [Steven Van Wel]; оно используется на условиях лицензии Creative Commons.

Создаем карту содержимого

Конечно же, имеется множество других приложений для создания небольших презентаций вроде этой, но наше позволяет исследовать доступные опции для введения в проект интерактивности.

Первым делом создадим «содержимое», чтобы точно знать, что происходит на каждом слайде. Для простоты у нас будут стартовая и завершающая страницы и три вопроса – каждый с экраном верного и неверного ответа. То есть всего 11 слайдов. Начнем с пустой презентации, хотя можете начать и с шаблона (или создать собственный).

> «Наш проект позволит исследовать некоторые доступные опции.»

Для добавления некой видимой структуры зададим имена слайдов в соответствии с их содержимым. Легче всего это сделать на вкладке Сортировщик слайдов. Щелкните правой кнопкой мыши на каждом из них по очереди и выберите Переименовать, чтобы задать новое имя.

Создание киоска

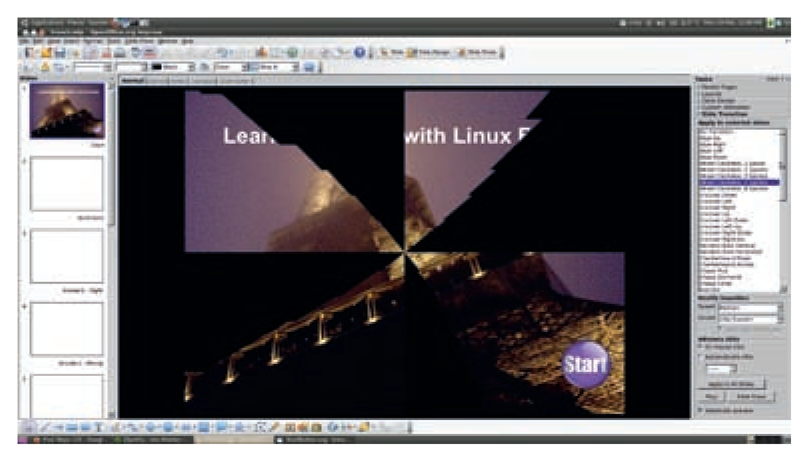
Если вы создаете презентацию для киоска – а именно, простые формы, между которыми переходит пользователь, вероятно, будет достаточно разработать и сохранить проект в виде обычного файла *OpenOffice.org*. Это даст вам несколько замечательных опций, отсутствующих при проигрывании Flash, который мы используем здесь. Например, можно применять переходы между слайдами, анимацию и текстовые эффекты. При выгрузке во Flash все это останется за бортом.

Разработку анимации в *Impress* мы опишем в следующих выпусках учебника «Первые шаги».

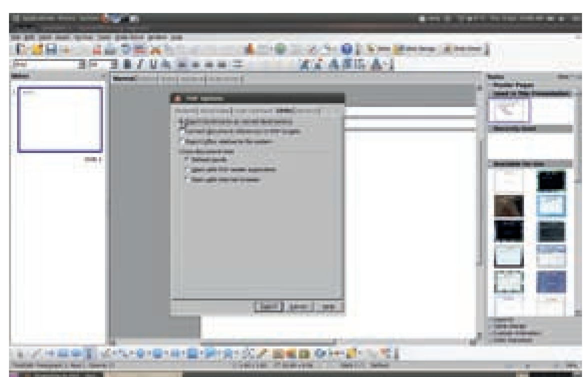
По завершении, начинайте добавлять на слайд дизайн. Например, можно поместить на главную страницу броское фоновое изображение, используя меню Вставка > Изображение > Из файла, а затем растянув его на весь слайд. Теперь, используя панель инструментов внизу основного окна, выберите иконку Текст, щелкните на слайде и создайте заголовок вашего проекта.

По умолчанию следующий слайд появляется после щелчка пользователя в любом месте предыдущего; дадим ему для щелчков нечто более осязаемое. Можно создать форму через панель инструментов у нижнего края окна (и даже добавить метку), использовать фрагмент текста или импортировать кнопку из такого приложения, как *GIMP* или *Inkscape*. Мы сделаем последнее, и применим для создания кнопки меню Вставка > Изображение > Из файла.

С помощью маркеров, появляющихся при щелчке на изображении, можно изменить его размер – используйте клавишу Shift, чтобы сохранить пропорции – а затем перенести его на подходящее место.



› В *Impress* достаточно много различных переходов, но Flash их не любит.

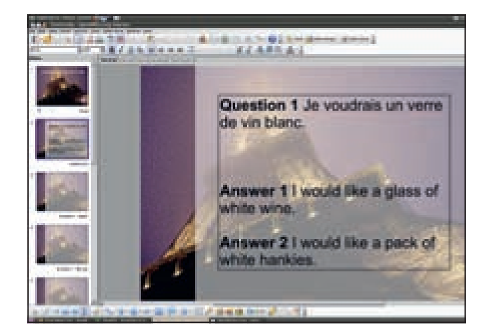


› Из *Impress* можно выполнять экспорт в PDF с закладками, но ссылки на слайдах не поддерживаются.

интерактив!

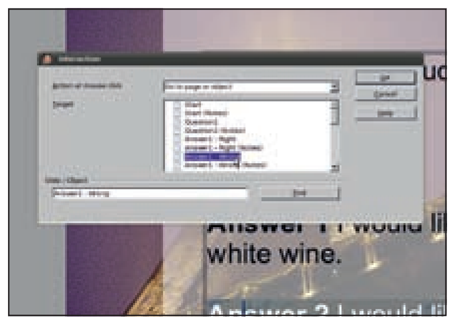


Шаг за шагом: Разработаем интерактивный опросник



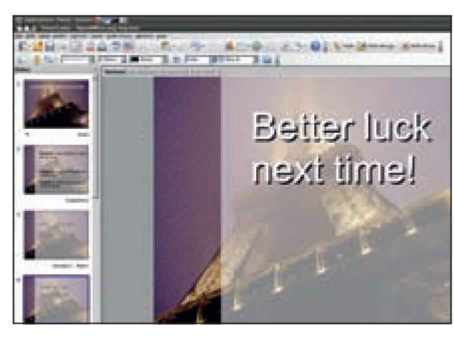
1 Создаем вопросы

Используя изображение и текстовый инструмент, имеющиеся в *Impress*, добавьте ваш первый вопрос на второй слайд. В нашем примере мы скопировали и вставили изображение с первого слайда, затем поместили сверху прозрачный белый квадрат для большей привлекательности. Текст – это просто текстовое поле, со словами, отформатированными при помощи панели форматирования в верху экрана.



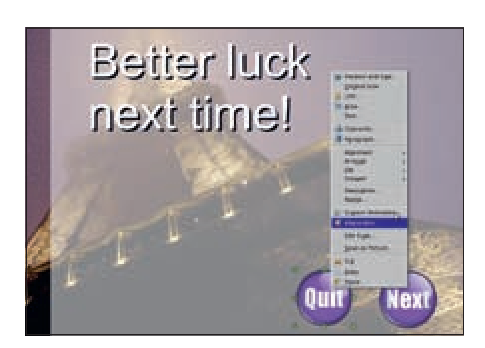
2 Добавляем выбор

Надо связать наши ответы с верным и неверным вариантами. Выделите один ответ, затем выберите Демонстрация > Действие по щелчку мыши... В выпадающем списке выберите Перейти к странице или объекту и прокрутите список имеющихся страниц, пока не увидите нужную. Выберите ее и нажмите OK.



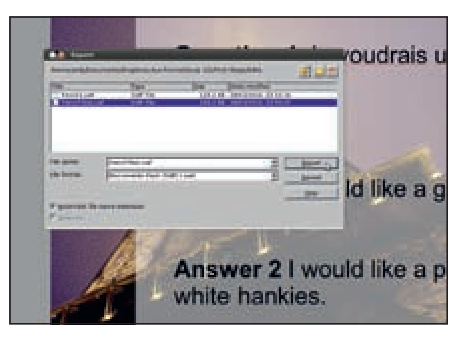
3 Создаем страницу с ответом

Так же, как и ранее, добавьте текст и изображения, чтобы показать пользователю, правильно он ответил или ошибся. Помните: чтобы все работало, вы должны проследить, что пользователи попадут на правильные страницы! Разработав страницы единожды, вы можете вырезать и вставлять их в качестве других страниц верных и неверных ответов.



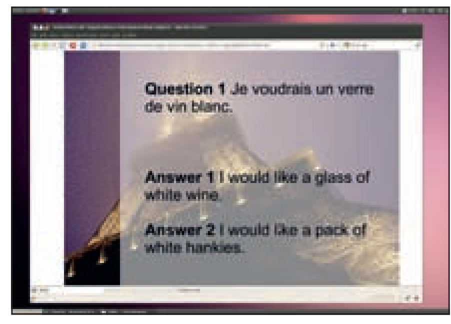
4 Ссылки Далее и Выход

Теперь вставьте изображение для пунктов Далее и Выход. Сделайте это, как ранее, и используйте Копировать и Вставить, чтобы добавить их на остальные страницы с ответами перед тем, как создать ссылки. Это позволит убедиться, что все они расположены в должном месте. Щелкните правой кнопкой мыши на первой кнопке Выход и выберите Действия по щелчку мыши. Определите ссылку для перехода на последнюю страницу, затем сделайте то же с кнопкой Далее, направив ссылку на следующий вопрос.



5 Выгрузка файла во Flash

После добавления всех опций можно вывести самодостаточный файл Flash. Его сможет использовать любой человек с установленным *Flash Player* в Windows, Linux или OS X. Вызовите Файл > Экспорт и в строке Формат файла выберите из списка Macromedia Flash (SWF). Введите имя файла в соответствующем поле и нажмите кнопку Экспорт.



6 Тестируем в браузере

Вы можете протестировать только что созданный проект, открыв *Firefox* и выполнив Файл > Открыть файл. Перейдите в место сохранения и выберите SWF-ролик. Опросник должен появляться в вашем браузере, а ссылки – вести в соответствующие места. В отличие от больших изображений, созданный нами 11-страничный опросник занимает всего 156 КБ, что превосходно даже для отправки друзьям по почте.

» Через месяц Мы создадим анимацию в *Impress* и эффекты с *Compiz*.

Firefox: В Сети

Список расширений для Firefox настолько длинен, что оторопь берет. **Джульетта Кемп** подстелит вам соломки, выбрав для вас лучшие.



Одним из плюсов *Firefox* является наличие расширений – частей кода, которые можно установить, чтобы ваш браузер работал именно так, как вы хотите. Полный список доступных расширений можно найти на https://addons.mozilla.org/firefox, но при их количестве где-то в районе 5000 непонятно, с чего начать; поэтому мы составили для вас список лучших.

Расширения просты в установке: большинство из них можно получить, выбрав в меню Инструменты > Дополнения и выполнив поиск во вкладке Поиск дополнений. После установки, как правило, нужно перезапустить *Firefox*, чтобы изменения вступили в силу.

Некоторые расширения предлагают параметры настройки, получить доступ к которым можно во вкладке Расширения. Здесь же вы можете удалить дополнение. После установки некоторых расширений в меню «Инструменты» появляется пункт, позволяющий изменять настройки. Если вам не нравится, как ведет себя расширение, стоит проверить его параметры и настройки, прежде чем удалить его совсем.

Расширения тестировались в *Firefox* версии 3.6 – это текущая стабильная версия; но в вашем дистрибутиве может быть другая. Большинство расширений должны работать с любой версией старше 3.0, или установите версию 3.6 с сайта Mozilla: www.mozilla-europe.org/ru/firefox.



Наш эксперт

Джульетта Кемп
Системный администратор и писатель, и она слишком много сидит в Интернете. Ей не забыть восторг от обретения вкладок, и теперь она открывает по 20 за раз.

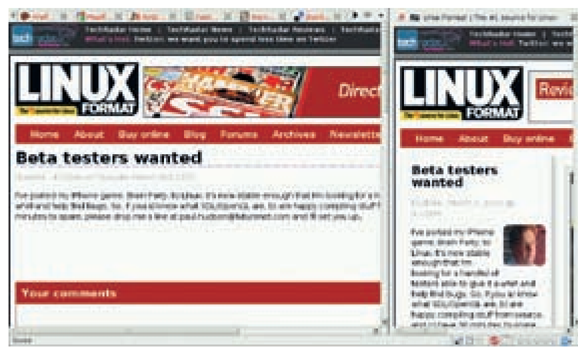
Часть 1 Изменяем облик вашего браузера

Fox Splitter

Первое, что мы рассмотрим – *Fox Splitter*: он позволит разделить экран вашего браузера на несколько окон меньшего размера, по вертикали или по горизонтали. Вы можете повторить этот шаг сколько угодно раз.

Понятно, что пригодится это обладателям большого экрана – на нетбуке тут, видимо, ловить нечего. Но при достаточном пространстве удобство будет очевидным, особенно если вы выполняете нечто вроде исследования и хотите выстроить несколько страниц рядом для сравнения; или – если вы пишете пост/комментарий в блог или сообщение в электронную почту и хотите иметь возможность проверить цитаты, не перескакивая между вкладками каждую пару секунд.

Приятной штукой здесь будет небольшая красная булавка, позволяющая синхронизировать прокрутки между экранами таким образом, что ведомая панель ведет себя аналогично главной. Это можно делать как по вертикали, так и по горизонтали. Другая полезная функция – поддержка drag-and-drop: перетащите ссылку на середину правой стороны текущей панели, и появится небольшое поле со стрелкой. Вставьте туда ссылку, и она будет открыта в новой панели в направлении, указанном стрелкой.



› ***Fox Splitter*** **позволяет удобно просматривать два сайта рядом друг с другом; можно также настроить синхронную прокрутку.**

Stylish

Stylish позволяет легко изменять стиль выбранных вами сайтов. Вы можете установить готовый стиль с сайта http://userstyles.org (или какого-нибудь другого), или написать свой собственный. Все они быстро устанавливаются и легко управляются с помощью значка S в нижнем правом углу строки состояния.

Сайт http://userstyles.org содержит более 23 000 доступных стилей. С учетом огромной популярности Facebook, YouTube и Google, возможно, неудивительно, что эти сайты появляются в списке чаще всего. Нам очень понравился Cleaned Up от YouTube, который счищает с YouTube некоторую накипь; также найдется много-много версий для YouTube с черным фоном. Hello Kitty от Facebook тоже ужасно популярен. Каждому свое...

с удобствами

Можно искать подходящие стили, применимые к определенным сайтам. Есть ряд доступных стилей *Guardian* и BBC, часто придающих сайту более минималистичный вид Немало стилей блокируют рекламу (подробно об этом далее). Мы также отыскали стиль Metafilter – он дает черно-синие цвета на белом фоне, гораздо легче воспринимаемые глазом, чем их обычный мутно-голубой.

К сожалению, http://userstyles.org страдает тем же недостатком, что и любой другой сайт, нацеленный на пользовательский контент: не все стили, доступные на сайте, на самом деле интересны, а то и не несут заявленный функционал. Кроме того, всегда есть риск того, что сайт изменит CSS-шаблон, который нарушит стиль, заданный пользователем.

Полезнее в этом плане, вероятно, интерфейс, предлагаемый *Stylish*: он поможет вам писать собственные стили для определенного сайта. Он позволяет легко задавать простые изменения (например, подправлять цвет фона) по собственному усмотрению на сайтах, где вы бываете регулярно. Имейте в виду, что с безопасностью пользовательских стилей могут возникнуть проблемы, если они содержат XBL-привязки (которые могут включать исполняемый код). Стили на http://userstyles.org не содержат XBL-привязок, но к тем, которые поставляются из других источников, следует отнестись с осторожностью.

Чудеса одним махом

FoxClocks показывает в нижней строке браузера время в различных частях мира. Это ценно, если у вас есть коллеги или друзья, проживающие в других часовых поясах, и вы бы не хотели разбудить их в 3 часа ночи.

Automatic Save Folder Укажите, где сохранять определенные типы файлов, на основе их имен.

Better Gmail 2 Добавляет различные возможности для работы с Gmail, в том числе иерархические метки и значок непрочитанных сообщений на вкладке вашего браузера. Это пакет скриптов Greasemonkey.

RightToClick Позволяет вызвать контекстное меню или выделить текст на странице, даже если сценарий JavaScript вам этого не разрешает.

SkipScreen Пропускает промежуточные страницы на некоторых сайтах – хранилищах файлов, таких как RapidDownload.

Firebug Обязателен для web-разработчиков. Позволит редактировать, контролировать и отлаживать HTML, CSS и JavaScript «вживую».

Часть 2 Облегчаем навигацию по сайтам

GleeBox

Многие считают, что сосредоточение работы в клавиатуре, не прибегая к содействию мыши, полезнее для спины и запястий. *GleeBox* расширяет клавиатурный функционал вашего браузера, который по умолчанию перекладывает на мышь достаточно много функций.

Это экспериментальное расширение, и вы не сможете добавить его прямо из окна дополнений. Чтобы разрешить установку, перейдите на страницу расширения (http://thegleebox.com/) и подтвердите серьезность своего намерения нажатием на кнопку Установить.

После установки и перезапуска *Firefox* найдите на текущей странице ссылку, которую вы хотите открыть. Нажмите G, чтобы вызвать поиск *GleeBox*, и набирайте имя ссылки. Когда она подцветится, нажмите Enter, чтобы открыть ее, или Shift + Enter, чтобы открыть ее в другой вкладке. Если совпадений несколько, перемещаться между ними можно по нажатию Tab.

В *GleeBox* можно также выполнять поиск по закладкам. Если вы посещаете указанные в закладках страницы особенно часто, вы, несомненно, сочтете это полезным. Активируйте данную возможность, установив соответствующий флажок на вкладке Функции [Features] в диалоге Опции [Options] в меню Инструменты > gleeBox.

GleeBox не ограничивается ссылками. Вы можете использовать его и для навигации по странице. Нажмите G, а затем введите ?**h**, чтобы выбрать все заголовки (уровней h1, h2 и h3). Перемещайтесь между ними при помощи Tab и Shift+Tab. Дополнительные возможности включают !**share t** для отправки текущей странице в Twitter (и другие сервисы – Gmail, Facebook, Digg и Delicious). Для открытия букмарклета с определенным именем наберите ![**bookmarkletname**]. Нам понравилось запускать таким образом Zap – он ликвидирует все форматирование на странице; отличный вариант для случаев, когда вы случайно забрели на сайт с ядовито-розовым текстом на канареечном фоне. (Чтобы изменить внешний вид сайта, который вы регулярно посещаете, поиграйте с опциями *Stylish*, как описано выше).

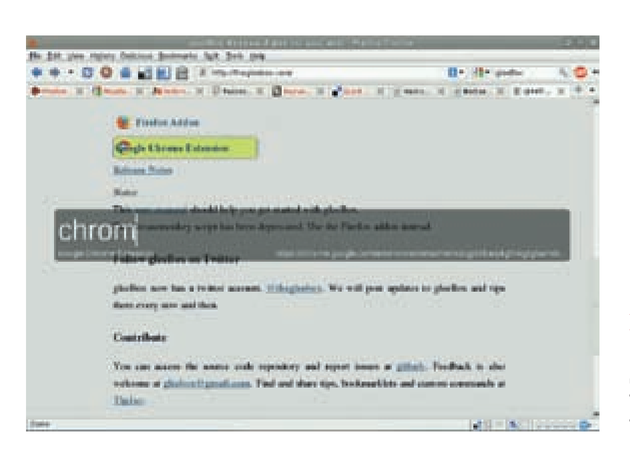
GleeBox также взаимодействует с YubNub, создающим «командную строку для web». Чтобы убедиться в этом, нажмите G и введите:

```
: xe -amount 100 -from USD -to RUB
```

Вы автоматически перейдете на сайт конвертора валют www.xe.com, который услужливо сообщит вам, сколько сегодня стоят 100 долларов в рублях. Для получения дополнительной информации, а также для создания своих собственных команд YubNub полезно будет просмотреть учебник *GleeBox* и сайт YubNub.

Скорая помощь

Если вы еще не применяли *Greasemonkey* – попробуйте: это поможет вам настроить «под себя» любую страницу через JavaScript. Более подробную информацию см. в **LFX118**.



› Благодаря расширению *GleeBox*, для навигации в Интернете достаточно только клавиатуры.

Скорая помощь

Нажатие кнопки Collections [Коллекции] на сайте Mozilla предоставит вам наборы дополнений, хорошо совместимых друг с другом.

FireGestures

FireGestures – это *GleeBox* навыворот: он снимает необходимость обращаться к клавиатуре, увеличивая число команд, доступных с помощью мыши. По умолчанию большинство команд использует Жесты мыши [Mouse Gestures] в понимании *FireGestures* (нажмите и удерживайте правую кнопку мыши и перемещайте курсор), но ряд других жестов, в т.ч. колесиком мыши, «коромыслом» (удерживая левую кнопку мыши, жмите правую, или наоборот) и мышью-плюс-клавиши, также поддерживается.

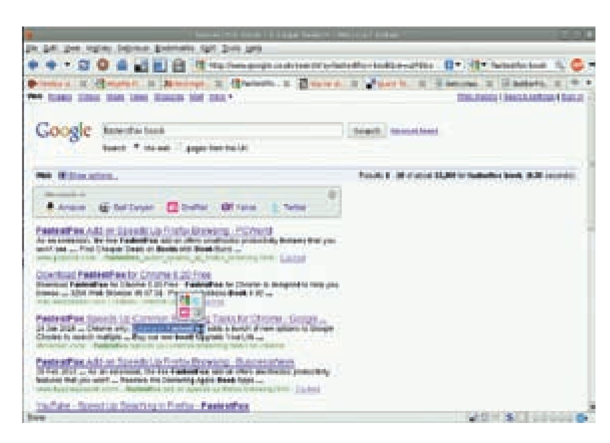
Нам особенно понравились жесты колесиком мыши над вкладками, (по умолчанию не включены). Они позволяют переключать вкладки, прокручивая колесико мыши на панели вкладок. А можно прокручивать колесико, придерживая правую кнопку, для просмотра списка всех вкладок, затем продолжив их листать. Единственный дефект этого варианта – сам список не прокручивается, и если вы открыли столько вкладок, что он вышел за нижнюю часть экрана (не то чтобы мы хотели, чтобы наши браузеры были в такой неразберихе...), здесь вы к ним доступа не получите.

По умолчанию, отслеживается траектория мыши; зеленый шлейф покажет вам, где, по мнению *FireGestures*, побывал ваш курсор, но, к счастью, это предусмотрено только для жестов мыши, а не всех движений подряд. Это преимущество по сравнению с другими расширениями для жестов мыши, применявшихся в прошлом; и его легко выключить, если оно вас раздражает.

FastestFox Plus

FastestFox Plus (ранее известное как *SmarterFox*) ускорит ваш браузер целой гаммой способов. Одной из функций, пользу от которой мы оценили, едва установив ее, является Endless Pages (Бесконечные страницы). Она обнаруживает, есть ли на данной странице ссылка Далее (такие бывают, скажем, внизу страницы блогов), загружает эту следующую страницу и замещает ею текущую, как только вы ее дочитаете. Это сильно сокращает трату часов на чтение архивов блога: теперь даже не нужно жать на ссылку, чтобы сделать это.

Другая функция – при выделении раздела web-страницы появляется небольшой виджет, дающий возможность отослать выбранный кусок в Twitter или поискать его на различных сервисах. По умолчанию это Wikipedia, Google или OneRiot (который индексирует ссылки, предоставленные в общий доступ на Digg, Twitter и других социальных сервисах обмена данными), но эти параметры можно изменить через меню Инструменты > FastestFox. При отправке выделенного текста виджет снабдит его и меткой, и ссылкой. Виджет особенно удобен для активных твиттеров и немало ускоряет поиск.



› В *FasterFox* есть окно Also Search On [Искать также в] и виджет поиска/отсылки в Twitter, отображаемый рядом с выделенным текстом.

FastestFox улучшает поиск в адресной строке, включая в выпадающее меню результаты поиска набираемого в Google, наряду с историей просмотра страниц, генерируемой Firefox автоматически. При обычном поиске в Google вам также предложат уточнения – нажав на такое, вы получите результаты согласно измененным критериям. Или можно искать **something book**, что дает массу ссылок на Amazon для **book**, но, увы, работает только для amazon.com, а не для европейских филиалов, и этого нельзя изменить.

Любой элемент *FastestFox* можно отключить из меню Инструменты, ограничившись набором нужных вам сервисов и убрав те, которые вас бесят.

Часть 3 Работаем с закладками

XMarks

Если вы используете *Firefox* на более чем одном компьютере и еще не установили *XMarks*, вы многое потеряли. *XMarks* позволяет синхронизировать закладки *Firefox* между двумя и более компьютерами. Просто установите дополнение, в ответ на запрос создайте учетную запись (или войдите в ту, что вас уже есть), и ваши закладки будут объединены в синхронизируемый набор.



› Боковая панель *Delicious* позволяет искать по вашим закладкам из *Firefox*.

Это довольно удобно, когда вы дома постоянно ловите себя на том, что ищете нужную закладку, а затем вспоминаете, что сохранили ее на своем рабочем компьютере; или если вам приходится передавать закладки между вашим ноутбуком и настольным компьютером вручную. Более того, оно позволяет синхронизироваться через различные браузеры, и теперь, кроме *Firefox*, поддерживает *Internet Explorer*, *Google Chrome* и *Safari*.

Если вы не хотите синхронизировать все закладки со своего домашнего компьютера с машиной в офисе, можно создать профили для ограничения состава закладок, синхронизируемых с другими компьютерами. Это позволит, например, не загромождать домашний компьютер вашими рабочими закладками.

XMarks имеет и другие, менее полезные функции, о которых все же стоит упомянуть: часто отправляемые в закладки сайты отмечаются флажком в результатах поиска Google, и нажатие на иконку в адресной строке выдает более подробную информацию о сайте и другие аналогичные ресурсы, которые могут представлять интерес для вас. Весьма удобный способ обнаружить в Интернете новую информацию.

Delicious Bookmarks

Вместо создания закладок на web-страницы в браузере, можно сохранить некоторые или большинство из них на web-сервисах вроде *Delicious*: они позволяют добавлять тэги и примечания к ним, а потом делиться ссылками с другими. Расширение *Delicious Bookmarks* облегчает работу с такими закладками.

Расширение устанавливает на панели инструментов три кнопки: одну для добавления текущей страницы в Delicious, другую – чтобы открывать боковую панель со всеми вашими закладками Delicious для поиска по ним, и еще одну, чтобы вызвать вашу страницу в Delicious. Есть также собственная панель инструментов Delicious, показывающая последние закладки, облака тэгов или избранные тэги. Теоретически, тэгами, облаками тэгов и избранными тэгами можно управлять с помощью расширения, но на практике обычно проще сделать это на главной странице Delicious.

Если вы – пользователь Delicious, расширение стоит установить уже ради одной кнопки tag-this-page [Присвоить странице тэги]: она вызывает всплывающее окно Save [Сохранить], через которое можно добавить собственный заголовок, заметки и тэги (есть и функция «предлагаемых тэгов»), выбрать сервис для отправки закладки или пометить закладку как приватную. В качестве бонуса прилагается возможность поиска в списке закладок из боковой панели.

Часть 4 Управляем браузером и приватностью

Flashblock

Flashblock давно уже является любимым расширением *Firefox* у многих пользователей, которых раздражает Flash-содержимое web-страниц. *Flashblock* просто заменяет любой элемент Flash пустым квадратом с кнопочкой Проиграть [Play] посередине. Хотите увидеть содержимое Flash – нажмите кнопку, и вперед. А не хотите – игнорируйте все это. Прямо как по волшебству.

Отдельные сайты можно исключить из блокирования – это полезно, если вы зачастили на страницы, без Flash бесполезные, и не хотите бесконечно нажимать на кнопку для отображения содержимого. *Flashblock* – один из тех отличных экземпляров кода, которые логичны и просто работают. Кроме того, вам больше не подсунут рикролл, что само по себе уже неплохо.

NoScript

Это еще одно простое расширение, которое отключает JavaScript, Java и другое исполняемое содержимое. Вы можете разрешить включать его с конкретных доменов, которым вы доверяете. Сперва вы обнаружите, что большинство посещаемых вами сайтов вызывают всплывающие окна в нижней части с предупреждением, что скрипты были заблокированы. Вы можете отменить в настройках отображение этого окна, предпочтя ему значок в строке состояния.

Есть простые варианты настройки разрешения запуска скриптов на сайте – временно или постоянно: бороздя просторы Интернета, вы заодно будете обучать *NoScript*, что такое хорошо и что такое плохо, и оно станет реже вмешиваться в вашу работу. Однако, если вы посвящаете много времени чтению новых сайтов, это может показаться вам утомительным.

Можно настроить и еще кое-какие вещи; но в конечном итоге это выбор между паранойей и удобством просмотра.

AdBlock Plus

У вас уже, вероятно, установлено это расширение, но мы все же его упомянем: вдруг кто-то на него еще не набрел. *AdBlock Plus* блокирует рекламу и баннеры всех видов.

Вы можете заблокировать что-нибудь вручную, щелкнув правой кнопкой мыши и выбрав AdBlock из контекстного меню; или просто подписаться на фильтр содержимого (при первом запуске или через меню Настройки [Preferences]), и большинство рекламных объявлений будет заблокировано по автоматически обновляемому списку. Можно также исключить из блокируемых определенные элементы, если ложное срабатывание создает вам проблемы.

Мы давно пользуемся *AdBlock Plus*; а на время написания данного учебника отключили его с целью убедиться, что оно все еще пригодно для рекомендации. Мы и обрадовались (оно, безусловно, работает!), и слегка ужаснулись тому, что творится без него (сколько же там назойливой рекламы!). Минут через пять мы снова его включили и облегченно вздохнули. Очень рекомендуем тем, кто, завидев рекламные объявления на web-страницах, готов треснуть кулаком по экрану.

BetterPrivacy

Многие сайты применяют cookies для Flash – это объекты, хранящие информацию для отслеживания пользователей и сведений о них. В отличие от обычных cookies браузера, Flash-cookies исключены из контроля конфиденциальности браузера.

> «Flash-cookies исключены из контроля конфиденциальности.»

Более того, даже если вы настроили свой браузер на удаление всех этих файлов при выключении, Flash-cookies не будут удалены. Вы можете изменить эти настройки через сайт Adobe, но, скажем так, без больших удобств.

BetterPrivacy исправляет это, и позволяет контролировать срок хранения Flash-cookies. Вы также можете просматривать Flash-cookies, уже сохраненные на компьютере. Нас несколько шокировал размер списка на нашей тестовой машине – некоторые из записей хранились аж с 2007 года. Удалив их полностью, мы потом не заметили каких-либо проблем.

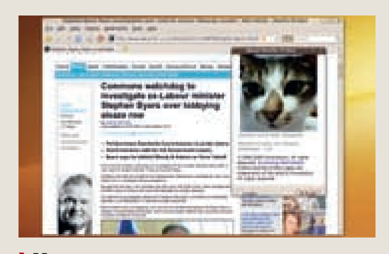
Учтите, что, как и с обычными cookies, если вы удаляете все при каждом закрытии браузера, вы можете потерять и нечто нужное. Чтобы это предотвратить, можно настроить исключения – полезная штука. *BetterPrivacy* является еще одним обязательным расширением, если вас хоть как-то волнует ваша конфиденциальность в Интернете. LXF

Игры и прочее

Consiousness Bell Воспроизводит звон тибетских чаш через случайные промежутки времени, напоминая вам о вдумчивости.

about:kittens Заменяет логотип *Firefox* в окне About случайным изображением котенка. Пристройте сюда свою киску!

Destroy the Web Мы уверены, что веселее ничего не придумаешь. *Destroy the Web* временно превращает любую web-страницу в игру-стрелялку. При щелчке на элементах они уничтожаются (под грохот взрыва!), а в конце подсчитываются набранные вами баллы.



› **Мы не утерпели и вставили котенка на сайт www.dailymail.co.uk**

POV-Ray: Строим

Трехмерное моделирование – это не только *Blender*. **Вячеслав Ястребцев** представляет другие рендереры, дающие новые возможности.



Наш эксперт

Вячеслав Ястребцев
Мечтает узнать, что мешает заниматься графикой в Linux, и исправить это.

Принято считать, что Linux обделён мультимедийными приложениями. До последнего времени это было обоснованным мнением, однако сегодня сообщество активно разрабатывает многочисленные программы для создания и просмотра мультимедиа. Трехмерные приложения – бесспорный лидер этого процесса. Стремительно развивающийся редактор трёхмерных сцен *Blender* (www.blender3d.org), используемый для создания трёхмерной графики и анимации профессионального качества; продвинутый рендерер *YafAray* (www.yafaray.org); наконец, вершина современных технологий визуализации – *LuxRender* (www.luxrender.net), обеспечивающий отрисовку изображений, практически неотличимых от фотографий – все они обещают скорый бум свободного ПО в медиа-индустрии. Но в тени стремительно растущих современных проектов тихо, без громких обещаний, развивается, пожалуй, самая почтенная система 3D-моделирования и визуализации, уходящая своими корнями в далёкие 80-е, но достойная пристального внимания даже сегодня. Имя этому аксакалу – *POV-Ray*.

Приготовимся к старту

Как и всякий старец, *POV-Ray* отличается консерватизмом и непростым характером. Двоичная версия релиза 3.6, доступная на www.povray.org, вышла аж в 2004 году, и её нормальная работа в современных дистрибутивах не гарантируется. Лучше сразу скачать исходный код *POV-Ray 3.7 Beta* и собрать его самостоятельно. Тут нас поджидают дополнительные хлопоты: при запуске конфигурационного скрипта требуется указать своё имя с помощью опции **COMPILED_BY=«имя»** (подойдёт любая комбинация букв и цифр). Кроме того, бета-версия требует постоянного ввода некого кода. Для его получения необходимо ввести в консоли команду **povray --betacode;** программа напечатает набор символов, который необходимо присвоить переменной POVRAY_BETA и далее экспортировать, выполнив **export POVRAY_BETA=код**.

Установив программу, настройте параметры рендеринга. Откройте файл **~/.povray/3.7/povray.ini** и добавьте в его конец две строки: **Pause_When_Done=On,** чтобы окно с готовым изображением не пропадало по завершении отрисовки, и **Output_File_Name=«pov_render.png»** – она указывает, в какой файл записывать созданное изображение.

Азбука SDL

Итак, всё готово для знакомства. *POV-Ray* не имеет встроенных средств интерактивного моделирования: есть множество программ, экспортирующих в его формат (в том числе альфа-версия *Blender 2.5*). Однако всю мощь *POV-Ray* можно раскрыть, только описывая сцену на SDL (Scene Description Language) – интерпретируемом языке программирования с С-подобным синтаксисом.

Сцены *POV-Ray* состоят из объектов, описываемых единым образом. Сам объект задаётся конструкцией вида:

```
тип_объекта {параметры}
```

Параметры бывают двух видов: обязательные и дополнительные. Обязательные необходимо указывать при создании объекта, сразу после открывающей фигурной скобки. Если забыть это сделать, *POV-Ray* сообщит об ошибке и аварийно завершит работу. Обязательные параметры могут быть числами или векторами, имеют фиксированный порядок и разделяются запятыми. Дополнительные параметры, напротив, требуют указания имени параметра, за которым следует его значение (число, вектор или объект). Разделять дополнительные параметры запятой не нужно.

Вектора в нотации *POV-Ray* записываются так:

```
<несколько чисел через запятую>
```

Примером вектора может служить **<1, 3, -2>** или **<0.2, 0.4, 0.9, 0.1>**. Вектора используются для обозначения координат и цветов.

Для ускорения работы *POV-Ray* позволяет использовать ряд сокращений: **x**, **y**, **z** – единичные вектора, совпадающие с соответствующими осями координат (**x** заменяет вектор **<1, 0, 0>**, **y** – **<0, 1, 0>**, **z** – **<0, 0, 1>**); при вводе дробей с нулевой целой частью можно не писать ноль перед точкой (**.1** вместо **0.1**).

Следует отметить различия координатной системы *POV-Ray* и *Blender*: в последнем ось z направлена в зенит, а вращение объектов происходит по часовой стрелке (если смотреть по направлению оси вращения). В POV-Ray ось z направлена к горизонту виртуального мира, а вращение происходит против часовой.

Вооружившись этими простыми правилами, приступим к написанию нашей первой сцены. Создайте текстовый файл с именем **sphere.pov**, откройте его в любом текстовом редакторе (*Vi*, *Emacs*, *Kate* «понимают» синтаксис SDL) и введите следующий текст:

```
camera {
		location <0, 0, -4>
		look_at <0,0,0>
		angle 50
}
light_source {
		<4,4,-3>, rgb 1
}
sphere {
		<0, 0, 0>, 1
		pigment {
				color rgb x
		}
}
```

В первой строке создаётся камера и определяются её основные характеристики: положение, отслеживаемая точка и угол поля зрения, соответственно. Положение камеры и точки обзора задаётся векторами. Как видите, ничего сверхъестественного нет – точно те же параметры (если не больше) мы бы указали, размещая камеру в *Blender*. В шестой строке создаётся источник света, которому требуются два обязательных параметра: вектор, задающий поло-

Лицензии, лицензии

Неприятный сюрприз: *POV-Ray* не является свободным ПО. Да, исходные тексты доступны, и вы можете найти программу в репозиториях своего дистрибутива; но в 1986 году, когда была начата работа над проектом, GPL еще не была столь популярна. За прошедшие 24 года над *POV-Ray* успело потрудиться множество людей, поэтому изменить лицензию на более приемлемую в современном Linux-мире, увы, не представляется возможным.

Новый мир



жение лампы, и цветовой вектор, задающий окраску и интенсивность света. На последнем стоит остановиться подробнее. Строго говоря, цвет в *POV-Ray* описывается пятью числами: привычной RGB-триадой основных цветов, пропусканием [transmit] и фильтрацией [filter]. Ключевое слово rgb подсказывает *POV-Ray*, что мы определяем только RGB-компоненты цвета, а пропускание и фильтрация будут нулевыми. Внимательный читатель может воскликнуть: «Цвет задаётся вектором, а в коде после rgb стоит число!» Ничего страшного в этом нет: *POV-Ray* догадается, что вы имели в виду, и заменит его на вектор требуемой размерности, все компоненты которого будут равны указанному числу. В нашем случае, получится вектор **<1, 1, 1>**, обозначающий чистый белый свет.

Наконец, в девятой строке создаётся сфера. Указываются два обязательных параметра: положение (вектор) и радиус (число), а далее идёт объявление нового объекта «пигмент» (pigment), вложенного в сферу. Пигменты задают цвет поверхности объекта-родителя: без них *POV-Ray* успешно отрисует сцену, но сфера будет выглядеть чёрным кругом. Внутри пигмента содержится параметр **color**. Здесь есть ещё одна причина для удивления: нам нужно задать цвет, а переменная **x** вроде бы описывает координаты? Все в порядке: x – это просто псевдоним для вектора <1, 0, 0>, что бы ни значили его компоненты; в данном случае они определяют красный цвет.

Созданную сцену необходимо скомпилировать (отрисовать). Откройте консоль, перейдите в каталог, содержащий файл **sphere.pov**, введите команду **povray sphere.pov**, и на экране появится изображение сферы. Вдоволь налюбовавшись на свое первое творение, щелкните по окну, чтобы закрыть его.

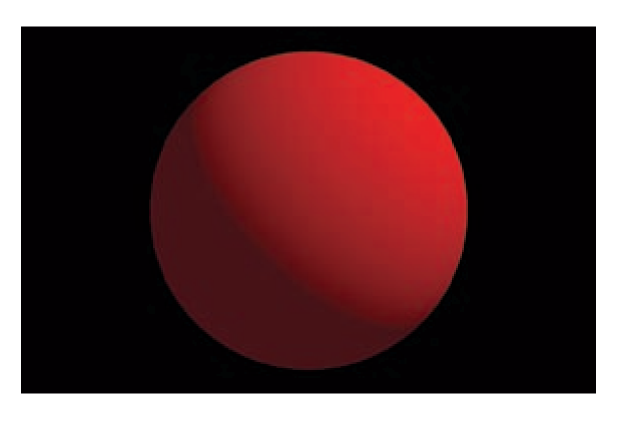
Мир без полигонов

Многие читатели, уже хорошо знакомые с трёхмерной графикой, могут заинтересоваться количеством полигонов в отрисованной сфере: уж больно гладкая у неё поверхность. Сообщаем: ни одного! Для описания сцены *POV-Ray* применяет математические функции, на ходу рассчитывая точки пересечения лучей света с идеально гладкими поверхностями их графиков, и хотя полигональные объекты можно использовать в сценах, они служат только для импорта моделей из сторонних приложений. С одной стороны, такой подход к моделированию непривычен, с другой – мы избавлены от многих «узких мест»: выбора между высоким качеством и объёмом занимаемой памяти, видимых изломов на поверхностях при достаточно малом расстоянии от камеры до объекта; наконец, сложные поверхности (например, горные хребты), требуют для корректного отображения просто чудовищного числа полигонов.

В *POV-Ray* существует два основных способа моделирования: с помощью объединения базовых форм (сфер, цилиндров, кубов и т.д.) в более сложные объекты, либо путем создания собственных функций, описывающих сложную поверхность. Первый подход удобно применять для разнообразных технических изделий – деталей механизмов, зданий; второй же идеален для создания ландшафтов. Сегодня мы подробнее остановимся на моделировании с помощью функций, а механикой позанимаемся в следующей части.

Небо и земля

Поставим себе задачу изобразить планету радиусом около 6000 условных километров, поднять на ней горы повыше Джомолунгмы, налить океан, прикрыть её атмосферой толщиной в десяток километров, а в довершение – вывести виртуальную камеру на орбиту и сфотографировать пейзаж. Не нужно бежать искать терабайтный винчестер – всё уместится в несколько килобайт!



› Проба пера: красная сфера на черном фоне в десяти с небольшим строках кода.

Для лучшего контроля за визуализацией нашей сцены, создадим новый файл с настройками рендерера. Просто скопируйте **planet.ini** с LXFDVD в директорию с вашим проектом – основные настройки *POV-Ray* в нем снабжены подробными комментариями; советую прочитать и их.

Некоторые величины, описывающие геологию планеты (радиус, высота гор и т.д.), нам понадобятся не раз, и лучше сразу дать им понятные имена, чтобы не запутаться. Для определения имён разнообразных объектов в *POV-Ray* используется оператор **#declare**. Допишите в файл **planet.pov** следующие строчки:

```
#declare atmoradius = 6020;
#declare planetradius = 6000;
#declare reliefheigth = 30;
#declare oceandeep = 11;
```

Как всегда, хорошим тоном будет сразу разнести основные компоненты нашей сцены (материалы, функции и сами объекты) по отдельным файлам. Скажем, создайте файлы **planet_texture.inc** и **planet_functions.inc**, а в **planet.pov** добавьте строки **#include «planet_functions.inc»** и **#include «planet_texture.inc»**. Первую из них придется продублировать и в начале файла **planet_texture.inc**. Ключевое слово **#include** велит *POV-Ray* перед дальнейшей обработкой сцены открыть и прочитать указанный файл.

Приступим к созданию объектов. Добавьте в **planet.pov** следующие строчки, создающие камеру и источник света:

```
camera {
        location z*-12000
        look_at 0
        angle 90
}
light_source {
        x*15000, rgb 1
        rotate y*40
}
```

Выражение **z*-12000** обозначает умножение вектора **z** на число -12000, т.е. перемещение камеры на 12000 единиц назад от центра мира (**z** – это вектор **<0, 0, 1>**, значит, **z*-12000 – <0, 0, -12000>**).

По умолчанию, в сцене *POV-Ray* присутствует рассеянное освещение для смягчения тени, но в космосе рассеянного света нет, поэтому нам следует отключить его:

»

```
global_settings {
	ambient_light 0
}
```

Теперь перейдите в файл **planet_functions.inc** – мы приступаем к созданию поверхности планеты. Сама она будет иметь форму шара, на котором располагаются микроскопические (относительно размеров планеты) неровности – горы. Для описания планеты необходимы минимум две функции: первая задает сферическую поверхность, вторая – создает рельеф. Нам же потребуется ещё одна: прибрежные области, как правило, имеют ровный, плоский рельеф, а в горах много провалов, трещин и изломов – третья функция будет отвечать за пересечённость местности.

Время кодировать

Начнём с создания сферы. Добавьте в **planet_functions.inc** строку:

```
#declare planetoid = function { sqrt( pow(x,2) + pow(y,2) +
pow(z,2) ) }
```

Она начинается с уже знакомого нам ключевого слова **#declare**, после которого идёт имя определяемого объекта. Слово **function** за знаком равенства создаёт новый объект-функцию, математическое выражение для которой приведено в фигурных скобках. Если у вас есть вопросы – обратитесь к полному листингу на **LXF**DVD; он снабжен подробными комментариями. Необходимо отметить, что переменные **x**, **y** и **z** внутри функций меняют своё поведение: теперь это не вектора единичной длины, а координаты точки в пространстве, для которой вычисляется значение функции. В нашем случае, значением функции будет расстояние от точки в пространстве до центра сцены, а множество равноудалённых от центра точек образуют сферу.

Чтобы сделать сферу видимой, нужно превратить абстрактную формулу в поверхность. Для этого применяется специальный объект – изоповерхность (isosurface), делающий видимыми области, в которых функция принимает определённое значение. Вернитесь в файл **planet.pov** и припишите к нему:

```
isosurface {
	function { planetoid(x,y,z) }
	threshold planetradius
	accuracy .000000001
	max_gradient 1.6
	contained_by { sphere { 0, planetradius+150 } }
	pigment { biosphere }
}
```

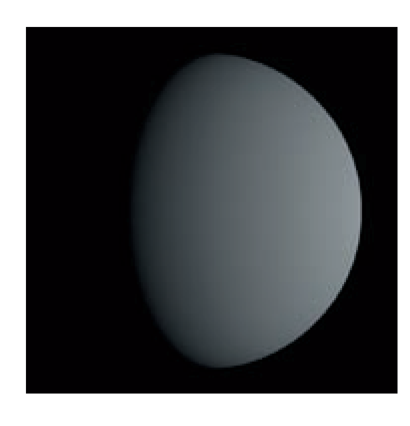
При создании объекта **isosurface** необходимо указать функцию, на основе которой будет строиться поверхность. У нас это **planetoid()**, определённая ранее в файле **planet_functions.inc**. Следующий параметр – **threshold** (порог), сообщающий, через область с каким значением функции пройдёт будущая поверхность. Мы используем здесь ранее заданную переменную **planetradius**, значение которой равно 6000 – это радиус сферы. Далее идут два важнейших для создания нормального изображения параметра: **accuracy** (точность) и **max_gradient** (максимальный градиент). Если их значения будут не оптимальны, мы получим чрезмерно долгую отрисовку с артефактами (тёмные полосы, дыры в поверхностях и т.д.). Легче всего настроить максимальный градиент: если он слишком мал или велик, *POV-Ray* напечатает в консоли предупреждение, предложив оптимальное значение. С **accuracy** сложнее: точность не должна быть ни слишком высокой, ни слишком низкой. Обычно требуются небольшие значения (порядка нескольких тысячных), а если очень малая **accuracy** не помогает избавится от артефактов, а, наоборот, усиливает их – это повод начать её увеличение. Параметр **contained_by** задаёт объект, ограничивающий пространство, в котором может располагаться изоповерхность. В нашей сцене ограничителем служит сфера с радиусом на 150 единиц большим, чем у планеты. Последний параметр назначает для нашей изоповерхности текстуру – это пигмент **biosphere**, который пока что определен в **planet_texture.inc** следующим образом:

```
#declare biosphere = pigment { color rgb .6 }
```

Посмотрите, что получилось: запустите в консоли **povray planet.ini** (или взгляните на рисунок). Да, это простой серый шар, но теперь мы можем деформировать поверхность, чтобы создать рельеф.

Скорая помощь

Иногда *POV-Ray 3.7 Beta* выдает сообщение об ошибке: «Parse Error: Redeclaring functions is not allowed - #undef the function first!». Создаваемая функция считается уже объявленной, что вызывает аварийное завершение работы. Проблема решается путём добавления команды **#undef <имя функции>** перед объявлением последней.



› Заготовка для нашей будущей планеты готова!

Поднимем горы

Вернитесь в файл **planet_functions.inc** и определите функцию, описывающую горы:

```
#declare highland = function {
	pattern {
		crackle
		warp {
			turbulence .4
			octaves 4
			lambda 2
			omega .7
		}
		scale planetradius*.025
	}
}
```

Здесь используется метод определения функции через встроенные в *POV-Ray* генераторы-паттерны [pattern], другими словами – процедурные текстуры. Функции, задаваемые с помощью паттернов, возвращают значения от 0 (чёрный цвет на текстуре) до 1 (белый цвет). В нашем случае используется паттерн **crackle** (аналог текстуры *voronoi* в *Blender*); слово **warp** определяет искажение базовой формы, **turbulence** – тип и силу искажения; **octaves, lambda** и **omega** — дополнительные параметры; **scale** – модификатор, изменяющий размер рисунка (паттерны, их модификаторы и процедурные текстуры будут детально рассмотрены в следующей статье, а самые нетерпеливые могут обратиться к комментариям в файле на диске).

Создадим еще одну функцию, описывающую контуры континентов и характер рельефа (*bozo* – близкий родственник *Blender*-текстуры по имени *clouds*):

```
#declare lowlands = function {
	pattern {
		bozo
		warp {
			turbulence 1
			octaves 4
			lambda 2
			omega .4
		}
		scale planetradius*.3
	}
}
```

Наконец, объединим две предыдущие функции в одну, описывающую весь рельеф планеты:

```
#declare landscape = function {(lowlands(x,y,z)+highland(x,y,z)*po
w(lowlands(x,y,z),4))*.7}
```

Осталось лишь применить созданную функцию к изоповерхности. Вернитесь в файл planet.pov и замените соответствующую строку на

```
function { planetoid(x,y,z) - landscape(x,y,z)*reliefheigth }
```

Снова запустите отрисовку командой **povray planet.ini**. Теперь она будет идти довольно долго, но вместо голого шара появится нечто, испещрённое горами и ущельями. Рельеф планеты готов!

Вдохнем жизнь

Как мёртвый каменный мир, наша планета неплохо смотрится уже сейчас, но куда интересней она будет выглядеть с атмосферой, водоёмами и лесами. Давайте нальем океаны – для этого нужно просто создать в **planet.pov** сферу со следующими настройками:

```
sphere {
    0, planetradius+oceandeep
    pigment { color rgb <0,.05,.5> }
}
```

Теперь перейдём в файл **planet_texture.inc** и как следует разукрасим нашу планету. На Земле вершины гор скрыты льдом, сами скалы лишены растительности и окрашены в красно-коричневые оттенки, равнины покрыты густой зеленью, а прибрежные зоны – песком или галькой; нам нужно воспроизвести это чередование цветов. Вначале зададим основные цвета, вписав перед определением пигмента **biosphere** строки:

```
#declare silt = color rgb .7;
#declare beach = color rgb <.5, .45, .05>;
#declare forests = color rgb <0, .2, .02>;
#declare rock = color rgb <.2, .08, .02>;
#declare ice = color rgb .8;
```

Далее надо изменить сам пигмент **biosphere** следующим образом:

```
#declare biosphere = pigment {
    function { landscape(x,y,z) }
    color_map {
        [0 color silt]
        [.36 color silt]
        [.365 color beach]
        [.37 color beach]
        [.38 color forests]
        [.48 color forests]
        [.54 color rock]
        [.55 color rock]
        [.56 color ice]
        [1 color ice]
    }
}
```

Здесь мы воспользовались объектом **color_map** чтобы изменять цвет поверхности в зависимости от значения функции **landscape()**.

Наконец, добавим атмосферу, определив в файле **planet.pov** объект **sphere** с параметрами:

```
sphere {
    0, atmoradius
    hollow
    material {atmosphere}
}
```

Атмосфера рассеивает и поглощает солнечный свет. В объявлении сферы использовано ключевое слово **hollow**, которое подготавливает объект к имитации объёмной среды, рассеивающей свет. Вместо пигмента задан материал **(material)** – сложнейший объект, объединяющий все оптические свойства предмета: от цвета поверхности до подповерхностного рассеивания света.

› **Нальем моря и пустим воздух. А заодно высадим лес.**

Перейдём в файл **planet_texture.inc** и добавим в нём еще один интересный материал, следующего вида:

```
#declare atmosphere = material {
    texture { pigment { color rgbt 1 } }
    interior {
        media {
            scattering { 3, rgb <.1,.8,1>*.003 }
            absorption rgb <.1,.8,1>*.003
        }
    }
}
```

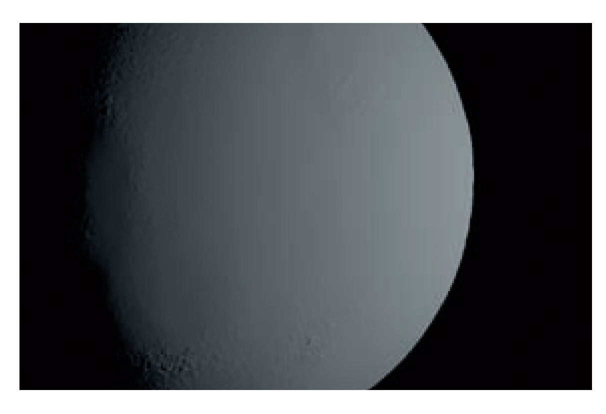
Запустив отрисовку, мы увидим голубоватую дымку, привычно окутывающую планету.

Запустим спутник

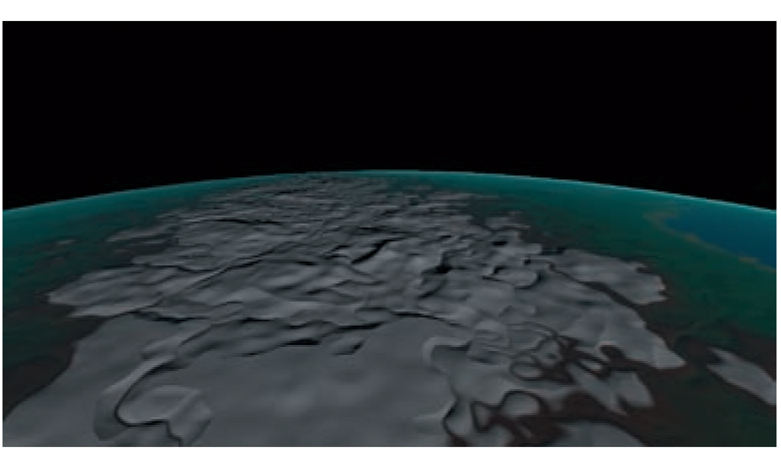
То, что у нас получилось – это не просто шар с текстурой! Замените старое описание камеры на следующее:

```
camera {
    #local camLoc = -6200 * z;
    location camLoc
    sky -x
    look_at camLoc - x + .3*z
    angle 90
    rotate <-26, -.8, 0>
}
```

Камера подойдёт ближе к поверхности планеты (по земным меркам, высота над поверхностью составит около 150 километров) и повернётся к одному из хребтов. Выполните отрисовку еще раз – и вы увидите горную цепь с ущельями, напоминающими русла рек. Подобные пейзажи можно найти по всей виртуальной планете: просто изменяйте вектор после слова **rotate**, чтобы переезжать с места на место. **LXF**



› **Лучше гор могут быть только горы, на которых еще не бывал...**



› **Земля в иллюминаторе! Ну, не совсем Земля...**

» **Через месяц** Моделирование и анимация механических частей и деталей.

libferris: Опыты

Хотите выгрузить изображение на Flickr или видеоролик на YouTube, не запуская браузер? С *libferris* и знаниями **Бена Мартина** это возможно.



Наш эксперт

Бен Мартин трудится над файловыми системами более 10 лет. После защиты докторской диссертации он занимается консультациями по вопросам работы с *libferris*, файловыми системами, поиском и программированием *Qt*/C++.

В последних версиях виртуальной файловой системы *libferris* сфера ее применения расширилась, включив сайты современных социальных сетей. Чтобы выгрузить свежий видеоролик на YouTube или снимок с мобильного телефона на Flickr, больше не нужен браузер. Ваша любимая программа для просмотра изображений также сможет загружать изображения с этих сайтов.

Сайты социальных сетей, фото- и видеохостинги в данной серии из двух статей я буду называть «web-сервисами», хотя это и термин из области распределенных вычислений: термин «социальная сеть», который теснее связан с Flickr, YouTube, Facebook и т.д., изрядно обвешан другими ассоциациями.

Есть масса программ, помогающих выгрузить контент на тот или иной web-сервис. Некоторые программы-просмотрщики также позволяют публиковать изображения на популярных онлайн-ресурсах. Однако реализация поддержки на уровне приложения имеет тот недостаток, что в каждой программе должна быть явная поддержка каждого web-сервиса. Вы неизбежно наткнетесь на утилиту, которая не сможет сохранить данные на очередном сайте. А вот бы, например, ваш сайт фотохостинга был просто файловой системой! Тогда можно было бы просмотреть его в любом редакторе, открыть в *GIMP*, изменить и снова сохранить в сети.

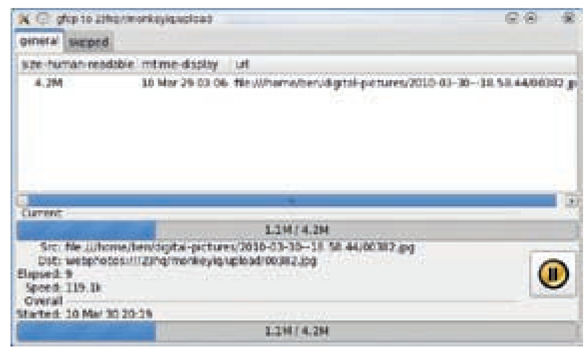
Понятие файловой системы может быть расширено от настольного компьютера до встраиваемых устройств – через сеть. Существует множество мелких утилит для мобильных телефонов, позволяющих публиковать фотографии онлайн. Будет это учетной записью Flickr или статусом Facebook, зависит от того, что позволяет программа. Производители телефонов стараются создавать партнерства с сайтами, таким образом вынуждая вас выбирать одного из их хостинг-провайдеров. Но будь эти сайты файловыми системами, вы бы просто копировали (командой *cp*) свежий ролик с вашего телефона через Bluetooth прямиком в Vimeo. Во всяком случае, выбор web-сервиса всегда должен оставаться за вами.

Армия разработчиков

К сожалению для пользователей этих web-сервисов, большинство провайдеров не облегчает разработчикам свободного ПО создание приложений для взаимодействия со своими службами, предоставляя только API Representational State Transfer (Rest), через который разработчик может получить список изображений, постов, видеороликов, комментариев и т.д. и выгрузить новое содержимое. Проблема обычно не в самом API, а в том, что web-сервис ожидает, что приложение аутентифицирует себя.

Для разных сайтов этот процесс немного отличается, но суть его в том, что для получения разрешения действовать от имени пользователя (выгружать данные и т.д.) программа должна пройти авторизацию на сайте. У каждого приложения есть ключ и некий общий с web-сервисом «секрет». Он требуется только на начальной фазе, когда приложение спрашивает сайт и пользователя, можно ли работать с вашим содержимым.

Проблема в том, что «секрет» должен оставаться секретом, а в приложении с открытым исходным кодом спрятать его негде. По сути, то же справедливо и для проприетарных программ – если кто-то захочет узнать пароль, используемый исполняемым файлом утилиты для выгрузки фотографий во Flickr, он сможет извлечь его уймой способов. Хотя пароль не передается по сети, он весьма предсказуемым образом используется перед тем, как у строки берется MD5. Так как приложение скорее всего применяет существующую реализацию MD5, легко прервать выполнение файла и вытащить из него пароль сразу после



› В *gfcp* можно перетащить изображения в список, и они будут загружены на www.23hq.com (как показано здесь).

с Web 2.0

появления (известного) кода для вычисления хэша. Процесс аутентификации приложения у web-сервиса приведен в документации разработчика Flickr.

К сожалению, в условиях использования web-сервисов обычно указывается, что разработчику нельзя раскрывать «секрет». Но не все потеряно. Приложение с открытым исходным кодом все еще прекрасно может пользоваться Flickr или YouTube, нужно только дать ему ключ приложения и пароль. Это как автомобиль без ключа зажигания – когда вы добудете ключ, он заведется.

Так каждый пользователь открытого ПО становится и «разработчиком». Он заходит на нужный web-сервис, регистрируется в системе как разработчик и получает ключ приложения и «секрет». Поместив их в файл, он может авторизовать созданное приложение (фактически – открытую утилиту, которую он скомпилировал и установил) у web-сервиса и начать им пользоваться. Условия использования сервиса для разработчика в этом случае могут приниматься или отвергаться каждым пользователем по его желанию. Конечно, для пользователя это лишнее неудобство, но это единственный способ распространения открытой программы, при котором условия использования сервиса не будут нарушены путем разглашения секретной информации.

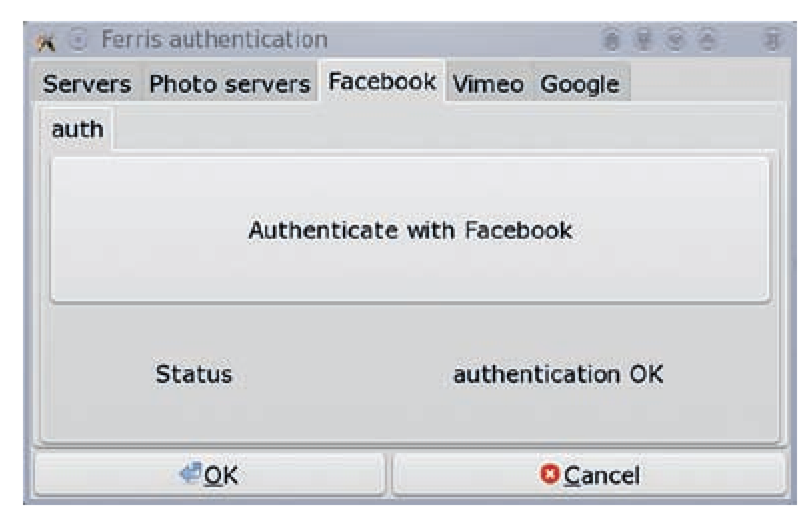
Обычно для коммерческих и бесплатных приложений предусмотрены различные условия использования web-сервисов. Так как файловую систему легко применить в коммерческих целях, выбирать условия использования сервиса должен пользователь.

Входим на сайт

Для Flickr получите ключ API и «секрет» по адресу www.flickr.com/services/api/keys/apply и запишите их в файлы **~/.ferris/flickr-api-key.txt** и **~/.ferris/flickr-api-secret.txt** соответственно. Ключи и «секреты» для всех web-сервисов хранятся в каталоге **~/.ferris** в отдельных файлах.

Для фотохостинга 23hq вы можете создать свои ключ и «пароль» самостоятельно. Их следует сохранить в файлах **23hq-api-key.txt** и **23hq-api-secret.txt**, соответственно. Эта политика может измениться в будущем, но поскольку 23hq использует Flickr API, для его ключа и «секрета» применяются те же самые файлы.

Чтобы настроить доступ к Facebook, зарегистрируйте «приложение» и поместите ключ API и «секрет» в файлы **facebook-api-key.txt** и **facebook-api-secret.txt**. Настройка доступа к Facebook – один из наиболее трудоемких процессов, потому что вместо простого получения ключа и «секрета» она напоминает написание небольшого web-приложения.



› Аутентификация в Facebook через *ferriscapplet-auth.*

Чтобы получить разрешение на выгрузку в Vimeo, нам понадобилось около недели – все может измениться, но по крайней мере готовьтесь подождать. Из врезки *Ресурсы* на последней странице урока вы узнаете, где добыть ключ и «секрет» для Vimeo; поместите их в файлы **vimeoapi-key.txt** и **vimeo-api-secret.txt** соответственно. Обратите внимание, что нужно явно отправить запрос на получение ключа через web-интерфейс.

Получение ключа и «секрета» – только полпути к монтированию сайтов. Далее запустите приложение *ferris-capplet-auth* (показанное выше), выберите web-сервис, в котором вы хотите авторизоваться с помощью *libferris*, и нажмите на кнопку Authenticate [Аутентификация]. Если ключ API в порядке, появится диалог с URL-адресом и кнопкой, позволяющей открыть его в *Firefox*. Сделав это, вы сможете сообщить web-сервису, что авторизуете взаимодействие с *libferris* (с вашими ключом API и паролем). Затем вернитесь в диалог *ferris-capplet-auth* и нажмите на кнопку OK, так что *libferris* свяжется с web-сервисом для завершения аутентификации.

После исчезновения диалога интерфейс *ferris-capplet-auth* должен показать вам статус ‘authentication OK’ для этого web-сервиса. Теперь мы готовы его смонтировать.

Для Google на данный момент *libferris* поддерживает только клиентский вход (Client Login). Пока *libferris* не получил поддержку дополнительного API аутентификации Google, нужно передавать ему имя пользователя и пароль напрямую.

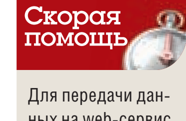
Эти забавные файлы

Итак, теперь мы готовы порезвиться с некоторыми сайтами. Ура! *Libferris* – виртуальная файловая система адресного пространства пользователя. Она во многом похожа на gnome-vfs (или gio) или KIO slaves в KDE. Можно либо пользоваться клонами *coreutils*, »

Не только для суперкомпов

Так как *libferris* поддерживает монтирование множества файловых систем, она может показаться новичкам, особенно тем, кто захочет скомпилировать ее из исходников, немного громоздкой. *Libferris* доступна в двоичных пакетах для последних версий Fedora, а также для устройств Maemo Diablo (N8x0). Часть из предлагаемых *libferris* вещей лучше всего выполняется на быстром процессоре с не менее чем 4 ГБ ОЗУ, а часть сработает и на более скромной системе с процессором ARM частотой 400 МГц и 128 МБ ОЗУ.

Скорая помощь



Для передачи данных на web-сервис используется каталог **upload**, а имя файла выдает сам web-сервис.

типа переделок *cp* – *ferriscp* и *gfcp*, либо смонтировать саму *libferris* с помощью Fuse и применять утилиты для работы с обычными файловыми системами.

Команда *ferriscp* работает в консоли как *cp*, а *gfcp* открывает графическое окно *GTK*. Обе имеют одинаковый набор ключей командной строки, хотя *ferriscp* молча игнорирует опции, относящиеся к графическому интерфейсу. Также существуют команды *ferrisls*, *ftouch*, *fcat*, *fmkdir* и т.д. – это клоны известных утилит командной строки с небольшими расширениями тут и там.

Видеофайл загружается на сайт таким образом:

```
$ gfcp \
~/my-video.avi \
youtube://upload/
```

А как насчет видео, которое вы сняли на свой смартфон? Если вы можете смонтировать телефон через Bluetooth с помощью *obexfs*, загрузка видеофайла на сайт делается так:

```
$ gfcp \
~/fuse/my-phone/.../latest.mp4 \
youtube://upload/
```

Хотите загрузить изображение на 23hq (или другой сайт, который использует Flickr API)? Вот вам набор команд:

```
$ alias 23hqu="ferriscp \
--target-directory \
23hq://me/upload"
$ 23hqu /tmp/foo.jpg
```

Команда *ferris-redirect* работает так же, как оператор перенаправления bash **>**. Зная, что последовательность | **ferrisredirect** эквивалентна **>**, вы сможете перенаправлять данные в любую файловую систему *libferris*. Укажите имя файла, в который нужно записать данные, и стандартный поток ввода *ferris-redirect* будет выведен в этот файл.

```
$ ftouch google://docs/datefileA.txt
$ date | ferris-redirect google://docs/datefileA.txt
$ fcat google://docs/datefileA.txt
Thu Aug 20 22:19:49 EST 2009
```

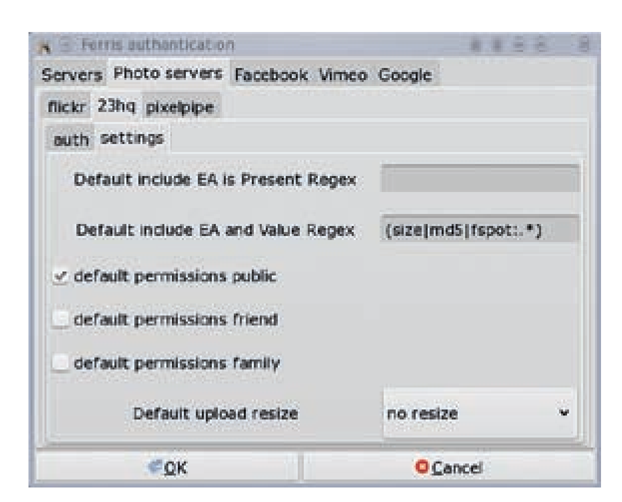
Скорая помощь

Общение с web-сервисами происходит по протоколу REST, потому что он очень прост и хорошо отражает то, как web-браузеры получают и отправляют информацию.

API Flickr

Учтите, что API Flickr поддерживается множеством сайтов, и любой из них должен быть доступен вам через *libferris*. Если интересный вам сайт с API Flickr «не дружит» с *libferris*, потребуется лишь создать производный класс от **WebPhotos** в **libferriswebphotos_shared.cpp** и возвратить URL-адрес вашего сайта и пути к ключу приложения и «секрету». Вы наверняка захотите расширить *ferris-capplet-auth*, чтобы аутентифицировать пользователей через графический интерфейс.

Реализация API Flickr началась в середине 2008 года с поддержки выгрузки фотографий. Это подразумевает возможность установки прав доступа по умолчанию через расширенные атрибуты (Extended Attributes), которые передаются посредством Flickr



› В окне *ferris-capplet-auth* есть набор опций для загрузки изображений на сайты, поддерживающие Flickr API.

API вместе со снимком. Интерфейс Extended Attribute (EA) файловой системы позволяет связывать с файлами произвольные метаданные в формате «ключ–значение».

В *libferris* концепция расширенных атрибутов виртуализирована, и они могут сохраняться на диске и считываться с него или получаться другими способами. Например, чтение атрибута MD5 заставит *libferris* вычислить MD5-хэш содержимого файла и вернуть его вам. *Libferris* может также перекрывать метаданные приложений; например, тэги, назначенные в *F-Spot*, являются для *libferris* такими же расширенными атрибутами.

Так как *libferris* поддерживает метаданные через расширенные атрибуты, можно заставить его скопировать некоторые из них на Flickr вместе с вашей фотографией. Как видно на экранном снимке вверху, я включил размер, хэш MD5 и все метаданные из программы *F-Spot* во все выгрузки на 23hq. Опубликовав фотографию, *libferris* записывает некоторые новые атрибуты в локальный файл изображения. Имя сайта, на который был выгружен снимок, сохраняется в *webphotoservice*, а идентификатор фотографии – в *webphoto-photo-id*. Таким образом, легко увидеть, выгружено ли изображение, и если да – получить его адрес на сайте.

gfcp

Команды, приведенные в листинге ниже, могут это прояснить. *Gfcp* копирует файл изображения в мою учетную запись Flickr. Поскольку я задал Default Include EA And Value Regex [Включать атрибуты по умолчанию] как '(size|md5)', я увижу тэги, показывающие размер и хэш MD5 исходного изображения на flickr.com. Вторая команда выводит некоторые метаданные локального файла изображения; как видите, *libferris* автоматически сохраняет идентификатор фотографии и имя web-сервиса в исходном файле фотографии. С помощью этих двух атрибутов мы получаем адрес изображения на Flickr – www.flickr.com/photos/monkeyiq/3857255129. Так что при просмотре локального файла изображения легко узнать, куда вы его загрузили.

Вот как изображение копируется на Flickr:

```
$ gfcp small-test-image.jpg flickr://me/upload
$ fls -lh --show-ea=name,size,webphoto-service,webphotophoto-id \
  small-test-image.jpg
small-test-image.jpg      683      flickr
3857255129
```

В выпадающем списке изменения размера по умолчанию в *ferris-capplet-auth* можно задать способ масштабирования

Экономим время на команды

Набирать **ferris-redirect** каждый раз при записи данных в файлы довольно нудно, но можно заставить *Bash* заменить это на более удобные команды перенаправления в *libferris*-совместимом тексте.

Для ввода *Bash* использует утилиту *readline*, у которой есть файл настройки, позволяющий заменить одну текстовую строку другой. Чтобы это заработало, добавьте следующие ниже строки в **~/.inputrc**. После этого в новых сеансах *Bash* для перенаправления данных в *ferris-redirect* вы сможете набрать только знак 'больше'.

Эти команды имитируют стандартные «дописать» и «усечь» и позволят перенаправить данные в любую файловую систему *libferris*.

```
$include /etc/inputrc
">>": "| ferris-redirect "
">|": "| ferris-redirect -T "
```

снимка перед его выгрузкой на сайт web-сервиса. Это удобно, если вы работаете с фотографиями с разрешением больше 10 Мпикс и хотите загрузить на сайт образец шириной 1280 пикселей вместо исходного файла размером 5 МБ. В *ferris-capplet-auth* можно задать только одно измерение, и оно определит ширину или высоту в зависимости от того, что больше у выгружаемого изображения. Например, если это значение равно 1024 и ширина больше, то высота автоматически изменится так, что соотношение сторон останется прежним, и загрузится файл с шириной 1024.

Me — это я!

Корень вашей файловой системы Flickr появится в подкаталоге с вашим именем пользователя. Чтобы обеспечить работу универсальных скриптов, директория **me** ссылается на каталог вашего пользователя, так что **flickr://me** и **flickr://monkeyiq** для моей учетной записи одинаковы. Содержимое каталога **me** показано ниже. Каталог **upload** передает все записанные в него файлы изображений Flickr API, и в результате они появляются на сайте. Каталог **contacts** позволяет вам увидеть своих друзей по учетной записи Flickr. В каталогах **photosets** и **not-in-any-photosets** находятся все ваши изображения, причем в последнем – снимки, не относящиеся ни к одной из категорий. В каталоге **recent** находятся недавно загруженные изображения, а в каталоге **favs** – изображения, которые вы пометили как избранные.

```
$ fls -0 flickr://me
by-id
contacts
favs
not-in-any-photosets
photosets
recent
upload
```

Каталог **by-id** позволяет найти изображение по его числовому идентификатору ID. Это последняя часть адреса, который вы видите в браузере. Например, в адресе www.flickr.com/photos/gromgull/3831809948 ID фотографии – 3831809948. В файле с именем, соответствующим идентификатору фотографии, в каталоге **by-id** можно найти метаданные снимка. Если прочесть этот файл, например, командой *cat*, вы получите версию изображения в самом большом доступном размере. Например, две следующие команды скачивают изображение и отображают его локально.

```
# This is us getting metadata and an image off Flickr
through the filesystem:
$ fls --xml flickr://me/by-id/3831809948
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<ferrisls>
<ferrisls mtime-display="70 Jan 1 10:00"
name="3831809948" title=""
url="webphotos:///flickr/.../3831809948"
webphoto-id="3831809948"/>
...
</ferrisls>
$ fcat flickr://me/by-id/3831809948 >| /tmp/out.jpg
$ gimp /tmp/out.jpg
$ fcat flickr://me/by-id/3831809948 | okular -
```

Просмотрев содержимое каталогов **favs**, **not-in-any-photosets** или **photosets**, вы увидите метаданные всех фотографий, которые вы пометили как избранные или поместили в фотоальбомы. Опять же, как и в случае с каталогом **by-id**, при чтении файла из каталога **favs** с web-сервиса будет загружена фотография с наибольшим из доступных разрешений. В своих фотоальбомах вы можете заметить, что у каждой фотографии есть каталог **<ID>_comments**. Комментариям к фотографии соответствуют виртуальные файлы в нем, как показано ниже. Для создания нового комментария просто запишите новый файл в каталог комментариев. Об имени файла не беспокойтесь, так как вы вряд ли сможете угадать нужное – *libferris* позволяет задать любое имя файла, не используемое в данный момент, и переименует файл, когда он получит корректное имя от API Flickr. Также можно записать данные в существующий файл с комментарием, как показано в последней команде. Это удобнее, если вы пользуетесь файловым менеджером и не хотите возиться с именами файлов, чтобы изменить комментарий.

```
#Создание и обновление комментариев к изображениям
на Flickr через файловую систему:
$ fls -0 flickr://me/not-in-any-photosets/443571910_
comments
7213593-443571910-72157621814094035 …
monkeyiq 09 Aug 4 11:48    four more
7213593-443571910-72157621814721241 …
monkeyiq 09 Aug 4 13:06
7213593-443571910-72157621814771773 …
monkeyiq 09 Aug 4 13:    test9
7213593-443571910-72157621888320960 ... monkeyiq 09 Jul
30 12:00 water dragon
$ echo what you looking at | ferris-redirect -T \
   flickr://me/not-in-any-photosets/443571910_comments/
new
$ echo test10 | ferris-redirect -T \
   flickr://me/not-in-any-photosets/443571910_
comments/7213593-443571910-72157621814771773
```

Если создать расширенный атрибут (EA) **tag:foo** у изображения в файловых системах **flickr://** или **23hq://**, *libferris* попытается задать тэг 'foo' для изображения с помощью API Flickr. Точно так же, если записать '0' в tag:foo, *libferris* попытается удалить тэг 'foo' у изображения через API Flickr.

Просмотр директории **flickr://me/contacts** немного озадачивает: у каждого друга здесь есть собственный каталог. В каталогах друзей находятся их фотоальбомы и избранные фотографии. Так можно узнать, какие снимки друзья считают интересными. Обратите внимание, что возможна загрузка видео с помощью API Flickr, если сервис позволяет это. Я смог загрузить на сайт Flickr несколько видеороликов.

В будущем планируется более плотная поддержка геотэггинга в *libferris*. Также было бы очень здорово запускать *rsync*, чтобы обновлять в API Flickr зеркало локальной файловой системы, но с этим пока есть проблемы. LXF

Скорая помощь

Хотя условия использования некоторых web-сервисов могут ограничивать ваши возможности, с помощью *libferris* удобно делиться фотографиями с друзьями из Flickr, которым недосуг просматривать снимки на вашей домашней странице.

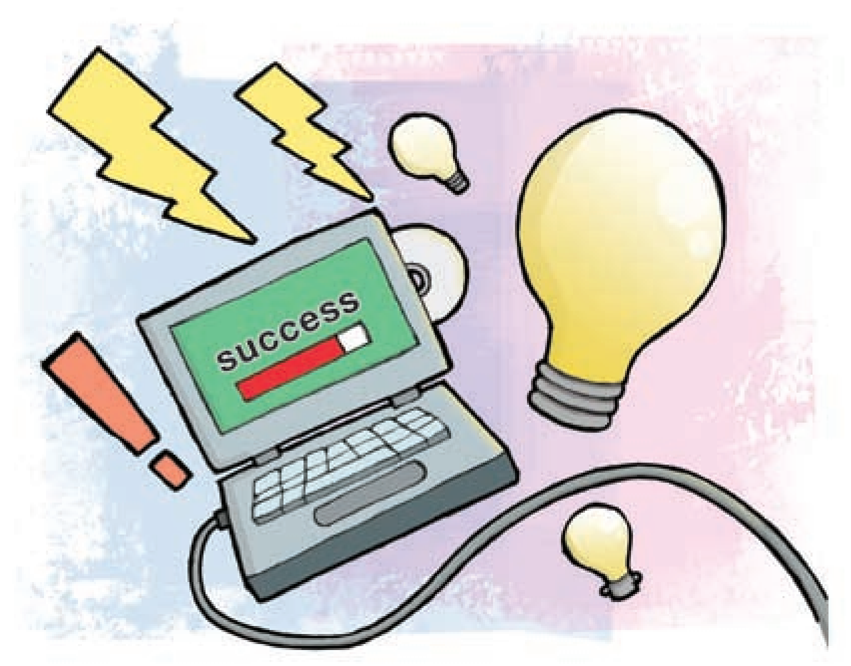
Ресурсы

- Геотэггинг в *libferris* www.linux.com/archive/feature/56674
- Документация разработчика Flickr www.flickr.com/services/api/auth.howto.desktop.html
- Получение ключа и «секрета»API Flickr www.flickr.com/services/api/keys/apply
- Ключ и «секрет» 23hq www.23hq.com/doc/api/faq www.23hq.com/doc/api/auth
- Ключ Youtube http://code.google.com/apis/youtube/dashboard
- Ключ Facebook http://wiki.developers.facebook.com/index.php/Connect/Setting_Up_Your_Site
- Ключ Vimeo www.vimeo.com/api/docs/desktopauth
- Клиентский вход Google http://code.google.com/apis/accounts/docs/AuthForInstalledApps.html
- Дополнительное API аутентификации Google http://code.google.com/apis/accounts/docs/GettingStarted.html

» **Через месяц** Смонтируем Facebook и YouTube и поработаем с Google Docs.

Base: Данные

Если вы не придумали, как бы поэффективнее управлять своей коллекцией DVD или списком контактов, **Боб Мосс** может показать вам, как это делается.





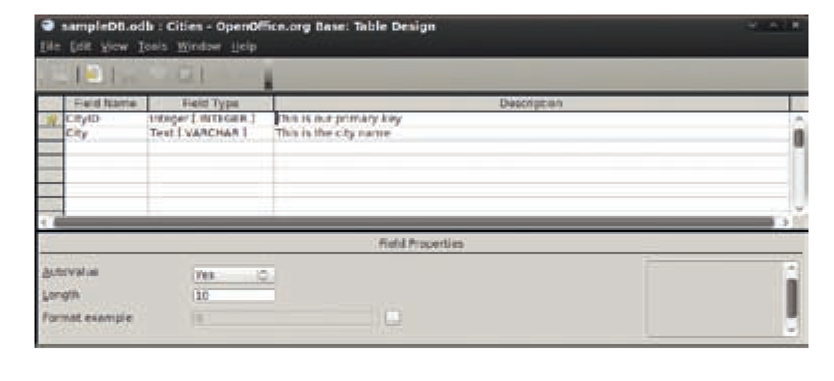
Наш эксперт

Боб Мосс делит свое время между программированием и обучением на факультете вычислительной техники.

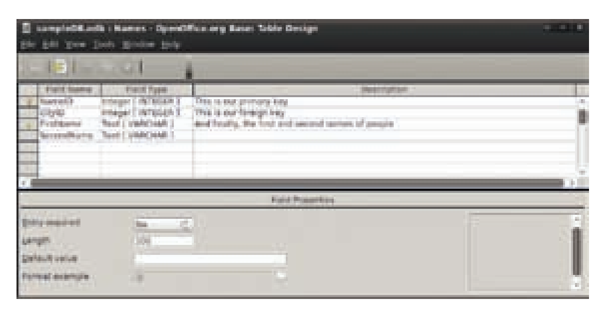
Вам уже, вероятно, знакома концепция электронных таблиц. Вы создаете набор столбцов с заголовками, задающими категории, и строки для перемещения по этому набору. Допустим, два столбца обозначают приход и расход, а строки относятся к различным датам. Ячейки на пересечении строк и столбцов содержат ваши финансовые данные, и вы в итоге получите баланс, по которому можно построить график.

А как узнать, кому вы платили? Или, что более важно, как управлять не только финансами? Здесь вам пригодится база данных. Базы данных являются скрытым механизмом большинства предприятий, и вы сталкиваетесь с ними каждый день в Интернете, так как они управляют самыми современными сайтами.

Не думайте, что разработка баз данных – это какая-то черная магия в исполнении хакеров из подземного бункера: в реальности их мощь доступна даже рядовому пользователю. Если вы взялись за наш урок, то вполне вероятно, что вы пока не собираетесь иметь дело с миллионами строк данных ради разработки корпоративной системы; поэтому процесс обучения будет не слишком сложен. Но даже приобретение навыка создания собственной базы данных для инвентаризации домашнего скарба, коллекции DVD или списка контактов может оказаться бесценным. Для целей урока мы предположим, что вы не знаете ровно ничего, и осветим только самое необходимое для ввода в тему.



› Создание таблицы названий городов — упражнение несложное.



› Здесь мы создаем таблицу имен людей и первичный ключ, а также внешний ключ для ссылки на таблицу городов.

Открытое решение

Те, кто сталкивался с базами данных ранее в *Microsoft Access*, сочтут *OpenOffice.org Base* более чем адекватной кросс-платформенной заменой. Большинство дистрибутивов не устанавливают эту часть *OpenOffice.org* по умолчанию, и вам придется сперва добавить ее через свой менеджер пакетов.

При первом запуске *OOo Base* вы увидите всплывающее окно с предложением создать базу данных. Вы можете спокойно выбрать вариант по умолчанию для создания пустой базы. Наблюдатели с орлиным взором заметят, что *OOo Base* напускает на вас мастеров, создающих примеры баз данных, но поскольку эти базы часто организованы не лучше обычных электронных таблиц, дважды подумайте, прежде чем их использовать.

Не пойдя на поводу у мастеров, вы получите совершенно пустую базу без таблиц. Создадим несколько таблиц, чтобы получить представление о работе базы данных. Они будут использовать первичные и внешние ключи; рекомендуем прочитать врезку на соседней странице, где изложена соответствующая теория.

В общем, имея пустую базу данных, создайте новую таблицу в Режиме дизайна – это чтобы мы могли задать столбцы в таблице

Имя [First name]	Фамилия [Second name]	Город [City]
Джон	Смит	Лондон
Джон	Смит	Ливерпуль
Том	Джонс	Лондон
Джо	Блоггс	Бристоль

в порядке



Города [Cities]	
CityID	City
0	Лондон
1	Бристоль
2	Манчестер
3	Ливерпуль

и типы данных для них, так же, как при работе с электронными таблицами. Наша первая таблица будет содержать названия городов, и нам понадобятся две колонки: CityID и City. Чтобы настроить CityID как первичный ключ, щелкните по нему правой кнопкой и выберите пункт «Первичный ключ». Неплохо также позаботиться, чтобы CityID было полем типа Integer (целое число), и включить для этого поля автоматическую нумерацию, чтобы при добавлении нового города не вводить первичный ключ (т. е. порядковый номер) вручную.

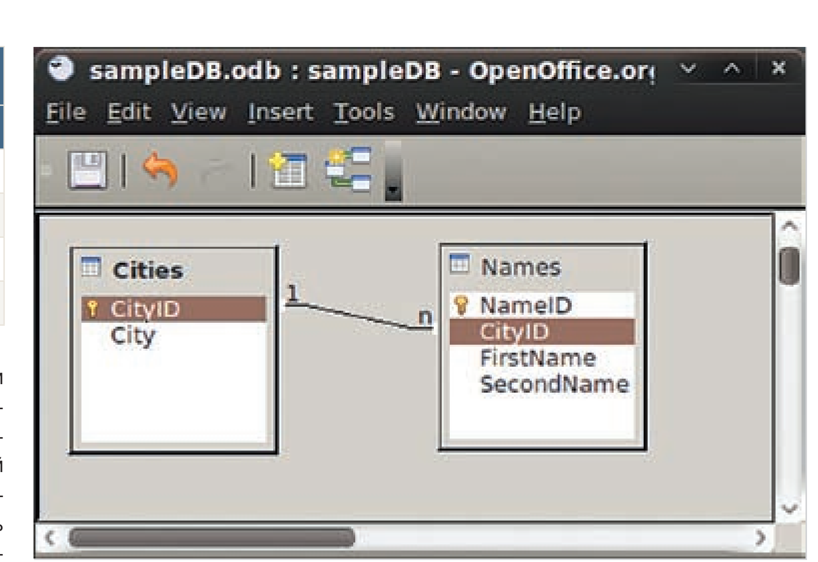
Сохраните это… вот вы и создали свою первую таблицу, для хранения названий городов. Чтобы ввести в нее данные, просто откройте таблицу с основной вкладки и добавляйте значения в поле City. Поле CityID будет заполняться автоматически при каждом добавлении новой записи. Чтобы удалить добавленные значения, щелкните правой кнопкой на самой левой части записи и выберите Удалить строки. Вы можете в любой момент вернуться в режим дизайна, щелкнув правой кнопкой в основном окне и выбрав Изменить.

> «Применим секретное оружие баз данных — запрос 'select'.»

Теперь сделаем нечто поумнее: свяжем людей с городами их проживания. В режиме дизайна, вам нужно будет создать новую таблицу с четырьмя полями. Одно назовем NameID – это поле типа Integer с автоматической нумерацией будет являться нашим первичным ключом. Следующая колонка получит имя CityID и будет тоже типа Integer. У нее не должно быть автоматической нумерации, так как это другой тип ключей и тут могут встречаться повторы. Также добавьте две текстовые колонки для имен [first name] и фамилий [second name]. Сохраните эту таблицу и откройте ее заново в стандартном просмотре для добавления новых записей.

Теперь свяжем CityID из таблицы Города [Cities] с CityID в таблице Люди [People]. Для этого перейдите на Сервис > Связи и добавьте туда обе таблицы. Затем для создания связи нужно щелкнуть и перетянуть CityID с первой таблицы на вторую. Вы должны увидеть **1** у таблицы Cities и **n** у таблицы Names, что показывает связь «один-со-многими».

Данные-то мы организовали, но запросто добавлять, просматривать или редактировать записи пока не получится. Для этого нужно создать наш первый запрос!



› Свяжем первичный ключ с внешним в редакторе связей.

Скорая помощь

Вы сможете найти обе базы данных из этого учебника на нашем диске.

Первичный или внешний?

В мире баз данных существует два типа ключей, о которых следует знать. Первый, который мы назвали «первичный ключ» – это поле ID (идентификатор), которое обязано быть в каждой одиночной таблице. Оно предоставляет уникальное значение ID для каждой записи.

Сперва это может показаться вам бессмысленным, но если в вашу таблицу со списком имен затесались два Джона Смита, движок базы данных по умолчанию сочтет их одним и тем же лицом. А назначив каждому из них разные значения ID, мы сообщим движку нашей базы, что это разные люди, хотя и полные тезки.

Люди [People]			
NameID	CityID	FirstName	SecondName
0	0	Джон	Смит
1	3	Джон	Смит
2	0	Том	Джонс
3	1	Джо	Блоггс

» Пропустили номер? Узнайте на с. 107, как получить его прямо сейчас.

Ответьте на мой запрос

Теперь применим секретное оружие баз данных – запрос 'select'. Мы уже немало поработали над организацией данных, но пока что таблицы все-таки отображают вещи более интуитивным для нас способом. Однако мы можем применить запрос select для сбора нужных данных из базы и отображения их в более дружелюбном к просмотру и редактированию виде.

Для этого просто щелкните на иконке Запросы в левой части окна базы данных и затем создайте новый запрос в режиме дизайна. Пользователям *Microsoft Access* это уже знакомо, а для остальных скажем, что нужно всего-навсего добавить в запрос обе таблицы. Далее добавьте поля First Name и Second Name в первые два столбца, используя соответствующие выпадающие списки, и отметьте поле City как третий столбец. Поля ID ни в какие столбцы добавлять незачем, так как нам нужно видеть содержимое базы, а не ее структуру.

Проделав это, сохраните свой запрос и затем запустите его двойным щелчком по нему в окне базы данных. Если вы все сделали правильно, сохраните и закройте вашу базу – для подготовки к следующему этапу...

Связываем таблицы

Следующий тип ключа – это «внешний [foreign] » ключ, который является более абстрактной идеей. Представьте себе две таблицы: одна содержит список имен, а другая – список городов. Понятно, что каждый человек может жить только в одном городе, тогда как в каждом городе живет много людей. В реляционной теории это известно как связь «один-ко-многим», и наш внешний ключ позволяет нам ее смоделировать.

В таблицу Люди мы добавили новый столбец, CityID, и для каждого человека указали в нем значение соответствующего первичного ключа города, где он живет. Мы допускаем наличие дубликатов в этом поле, поскольку знаем, что в каждом городе может проживать более одного человека. Затем мы связываем поле CityID в этой таблице с первичным ключом в таблице City.

В данном случае NameID в таблице Names и CityID в таблице City – это наши первичные ключи, а поле CityID в таблице Names – внешний ключ. Вы заметите, что у нас есть два Джона Смита, но они живут в разных городах. Один из Джонов Смитов и Том Джонс живут в Лондоне, так как значения их полей внешнего ключа одинаковы.

Применим это на практике

Итак, вы создали простую базу данных, содержащую две взаимосвязанных таблицы, и запрос, который состряпал из этих данных нечто полезное. Теперь применим полученные знания к реальному проекту, который вы сможете использовать в повседневных задачах. Для примера организуем вашу персональную коллекцию DVD.

Первым делом определим сведения, которые нужно хранить о каждом DVD. В нашем сценарии мы должны учесть, просмотрен фильм или нет, и находится ли DVD в вашей видеотеке или вы одолжили его другу. Также, возможно, вы захотите видеть информацию о самом фильме на диске, чтобы можно было сразу сказать, есть ли он в вашей коллекции или надо сгонять за ним в ближайший видеопрокат.

Далее обдумаем ключевые связи в этой базе данных. На одном DVD по идее может быть только один фильм [в России бывает и не так, но этот случай мы не рассматриваем, – *прим. ред.*], но у вас может найтись не одна копия этого фильма. Значит, в таблице DVD нужен внешний ключ, так как название фильма будет нашими повторяющимися данными. Мы можем также задать в таблице, просмотрен фильм или нет – для любого фильма это происходит только один раз, сколько бы ни было у вас DVD с его копиями (например, если вы видели фильм *Крепкий орешек*, это касается всех имеющихся у вас копий). Можно также задать в таблице DVD, есть ли фильм в наличии у вас дома: скажем, вы одолжили кому-нибудь диск с *Криминальным чтивом*, но он не последний – у вас остались другие его копии.

Итак, нам снова понадобятся две таблицы: одна будет содержать данные обо всех имеющихся у вас DVD, а другая – информацию о каждом фильме. Они будут выглядеть как таблицы, приведенные вверху и внизу.

DVDs

Ключ?	Имя столбца	Тип данных	Описание
PK [Первичный]	DVDID	Integer	Наш первичный ключ (сделайте ему автонумерацию)
FK [Внешний]	FilmID	Integer	Внешний ключ фильма
	LendStatus	Boolean	Отдан другу? Да/Нет

› Это готовый запрос select; его можно расширить с помощью параметров.

Поклон Джорджу Булю

Вы заметите в этих таблицах новый тип данных – Boolean. Он гарантирует нам, что на вопрос, находится ли DVD у нас дома или

Films

Ключ?	Имя столбца	Тип данных	Описание
PK	FilmID	Integer	Наш первичный ключ (сделайте ему автонумерацию)
	FilmName	Text	Да, вы угадали: это название фильма
	SeenIt	Boolean	Просмотрен? Да/Нет

Роли [Roles]			
Ключ?	Имя колонки	Тип данных	Описание
PK	RoleID	Integer	Наш первичный ключ
FK	FilmID	Integer	Внешний ключ к таблице фильмов
FK	ActorID	Integer	Внешний ключ к таблице актеров
	RoleName	Text	Имя персонажа

был ли просмотрен фильм, мы можем иметь только ответ: Да или Нет. И опять, легко сделать запрос select, который достанет нам эту информацию и отобразит в дружественном виде, поэтому задайте нужные связи в Сервис > Связи, как было показано в предыдущем примере.

А как с актерами?

Это следующий шаг, который можно проделать с вашей базой коллекции DVD, так как вы, вероятно, пожелаете выполнять поиск по актеру или режиссеру – их имени, биографическим данным и т. п. Задача более чем практическая, но мы столкнемся с весьма интересной проблемой. В фильме играет более одного актера, но актер мог сниматься более чем в одном фильме. То есть нельзя просто создать таблицу Актеры [Actors] и добавить внешний ключ FilmID, поскольку в этой таблице потребуется более чем одна запись для одного и того же актера. Это называется связью «многие-ко-многим», а такое базы данных не очень-то любят. Они работают со связью «один-ко-многим», и если мы хотим создать эффективную базу данных, нужно решить эту проблему.

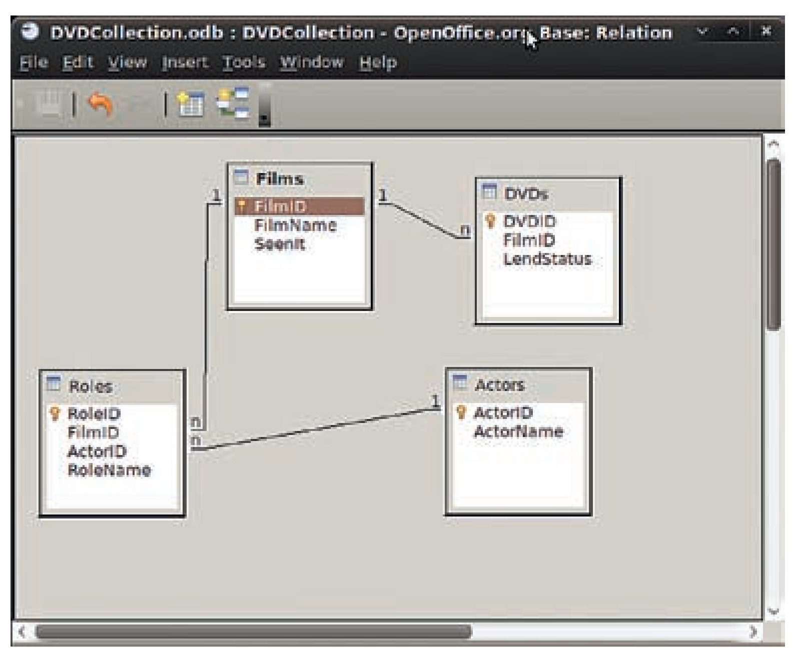
Чтобы ободрить вас, скажу, что не все потеряно. Ситуацию разрулит добавление таблицы Роли [Roles]. Мы знаем, что для роли маловероятно появиться дважды и что в фильме имеется более чем одна роль. Но роль (обычно) играется только одним актером, поэтому создание такой таблицы дает нам элегантное решение. Процесс, который мы только что описали, разработчики баз данных называют «нормализацией», и полное пояснение, как это работает и зачем мы это сделали, выходит за рамки данного урока.

Итак, чтобы ввести наше решение в действие, нужны две дополнительные таблицы: Роли [Roles] (вверху) и Актеры [Actors] (внизу).

Наконец, для завершения структуры базы данных свяжите первичные ключи со внешними в редакторе связей, как показано на рисунке выше, и можете начать заносить информацию о вашей коллекции фильмов.

Это может показаться неочевидным (и даже бессмысленным, если у вас всего-то и есть, что DVD с *Терминатором* и Арнольдом Шварценеггером), но эта база данных богата возможностями.

Актеры [Actors]			
Ключ?	Имя колонки	Тип данных	Описание
PK	ActorID	Integer	Это наш первичный ключ
	ActorName	Text	Имя актера



› Мы смогли разрешить наш конфликт «многие-ко-многим» между фильмами и актерами, введя понятие ролей.

С ее помощью можно поискать, какой DVD вы одолжили, задав соответствующие параметры в запросе. Она может даже предложить вам DVD для просмотра, основываясь на фильмах, которые вы уже видели. Или вы можете создать ряд форм (об этом мы на уроке не говорили), чтобы сделать ввод и вывод данных более дружелюбным. Еще лучше будет предусмотреть импорт информации прямо с web-сервиса вроде IMDb. Вы можете найти созданные базы данных на LXFDVD этого месяца, и вам будет легко приспособить их под ваши нужды или расширить за пределы данного урока. Возможности безграничны, и если вы сделаете нечто особо полезное, поделитесь с нами через forum.liuxformat.ru. LXF

> «Вам будет легко приспособить эти базы под любые нужды.»

Когда не хватает select

Запросы умеют не только выбирать данные и отображать их: манипулировать сохраненными в базе данными можно разными способами. Запросы могут обновлять существующие данные, создавать новые записи или удалять их; также бывают запросы, объединяющие таблицы и результаты других запросов. Короче говоря, есть масса действий, выполняемых над вашими данными в реальном времени, и они делают базы данных действительно очень мощными и универсальными.

Ограничением *OOo Base* является отсутствие соответствующих графических инструментов, что дезориентирует пользователей *Microsoft Access*, давно уже имеющих эту функциональность. Пользователи *OOo Base* лишены такого счастья и вынуждены писать запросы вручную в их SQL-представлении.

А что такое SQL? Это сокращение от «Structured Query Language» [Язык структурированных запросов]; его можно использовать для выполнения запросов прямо через движок базы данных, без графических оболочек. Как правило, это прерогатива разработчиков баз данных, поскольку запросы могут быть очень сложными, и одной неправильной строчкой легко угробить всю информацию.

Существует еще полный кошмар декартова произведения множеств, когда вы пытаетесь объединить две таблицы в одну новую. Для маленьких таблиц все получится хорошо, но в реальных задачах с относительно большими таблицами сервер при выборке сильно затормозит, а то и рухнет; так что лучше этого избегать.

Если вы пошли путем создания собственных SQL-запросов, сперва тестируйте их на похожих ненужных данных, чтобы избежать потерь. Надо ли напоминать, что следует делать резервные копии?

OOo Calc: Ваши

Любимая электронная таблица не желает округлять числа до нужного числа знаков? **Леонид Алифанов** и **Александр Маджугин** научат ее этому и многим другим трюкам!



Многие виды деятельности (инженерные расчеты, статистика, финансы и т.д.) сопряжены с необходимостью периодически выполнять однотипные расчеты высокой сложности. Электронные таблицы – один из самых удобных и распространенных инструментов для решения данной задачи. Этим уроком мы открываем цикл статей, посвященных деталям создания сложных расчетных шаблонов и использования функций Basic для расширения функциональности *Calc*.

Наш сегодняшний урок затрагивает самые базовые вещи и ориентирован в первую очередь на начинающих. В частности, мы рассмотрим пользовательские функции и их применение в прикладных расчетах в электронных таблицах.

Функции пользователя

При проведении сложных специализированных вычислений встроенных возможностей *Calc* зачастую оказывается недостаточно, или конвейеры, построенные из стандартных функций, получаются чересчур сложными. В этом случае на помощь приходит язык для расширения *OpenOffice.org* – StarBasic, позволяющий создавать пользовательские функции, в том числе и для электронных таблиц.

Функции StarBasic универсальны – они могут возвращать не только числовые значения, но и многострочные текстовые отчеты, а также массивы смешанных данных. То же можно сказать и о переменных, которыми оперируют эти функции.



Рис. 1. Пользовательская функция в таблице Calc.

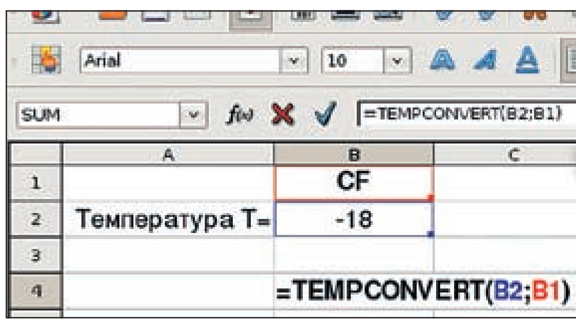


Рис. 2. Пример использования функции с двумя переменными.

Почему функции предпочтительней макросов, обрабатывающих данные, например, по нажатию кнопки? Во-первых, привычностью: пользователь электронной таблицы уже знаком с данной концепцией. Во-вторых, универсальностью: простой, однажды написанный код на Basic, не содержащий фрагментов, взаимодействующих с ячейками рабочей книги и не использующий специфических API, почти не зависит от изменений в языке или библиотек и может быть использован в любом компиляторе или интерпретаторе Basic. При необходимости, его несложно адаптировать для *Excel* и потом перейти обратно.

Пожалуй, достаточно вступительных слов – давайте рассмотрим пример простейшей функции:

```
Function HelloWord As String
        HelloWord = "Hello Word!"
End Function
```

Она вообще не имеет никаких входящих параметров и просто возвращает строку «Hello Word!». Чтобы воспользоваться ею в *Calc*, достаточно вписать в ячейку **=HelloWord()**, как и в случае со встроенными функциями (рис. 1).

Однако в реальности функции обычно принимают какие-то параметры (скажем, диапазон ячеек, подлежащих обработке). Реализуем это – напишем функцию, принимающую одно числовое значение (сумму) и возвращающую его же, но с учетом НДС:

```
Function NDS (ByVal clnSumm As Double) As Double
        Const cStavka = 18
        NDS = clnSumm/(100 + cStavka)
        NDS = NDS * cStavka
End Function
```

В этом случае синтаксис вызова функции также не отличается от традиционного: набрав **=NDS(B3)**, вы получите сумму с учетом НДС для числа, содержащегося в ячейке B3. Нужно больше входящих переменных? Просто перечислите их все в скобках через запятую. Пример такой функции можно найти на диске в файле **Листинг 1**, а ее использование проиллюстрировано на рис. 2.

Можно заметить, что в листинге 1 присутствуют две функции – **TempConvert()** и **Round()**. Первая из них вызывается из *Calc*, а вторая

Функции



необходима для её вычисления. Функция округления **Round(x,a)** есть в VBA для *Excel*, но отсутствует в *OpenOffice.org Basic*, поэтому нам пришлось написать свою. Функция **TempConvert()** не типизирована – по умолчанию ей присваивается тип Variant (универсальный тип, который может принимать любое значение). К слову, переменные можно вообще не объявлять: Basic сам будет распознавать их и возвращать значения по смыслу написанного вами кода. Но такой подход не вполне профессионален и в некоторых случаях может приводить к ошибкам или уменьшать быстродействие.

Изменим функцию **TempConvert()** так, чтобы она выдавала не значение, а текстовый отчет. Просто добавим в конце строки:

```
TempConvert="Температура воздуха " & Str(TempConvert) & "°."
```

Неизменяемый текст берется в кавычки, символ **&** обозначает сцепку; **Str()** преобразует число в текст. Естественно, результат можно было сделать и вариантным, с разными пояснениями при разных значениях ключа.

Массивы

А что если вам потребуется передать в функцию произвольный диапазон значений, как это делается, например, в стандартной **SUM()**? В данном случае соответствующую входящую переменную следует объявить как Variant. При этом необходимый диапазон может быть задан как обычно (текстом через двоеточие) или выделен при вводе с помощью мыши.

	A	B	C	D
1	1	8		
2	2	16		
3	4	1,5		
4			3R x 2C	
5	=N_POWER(A1:B3;4)			

	A	B	C	D
1	1	8		
2	2	16		
3	4	1,5		
4				
5	1	4096		
6	16	65536		
7	256	5,06		
8				

› Рис. 3. Результат вычисления функция массива занимает заданную область на листе рабочей книги.

	A	B	C	D	E	F	G	H	I	J	K
1											
2	STOCK1								STOCK2		
3	<DATE>	<TIME>	<OPEN>	<LOW>	<HIGH>	<CLOSE>	<VOL>		<DATE>	<TIME>	<OPEN>
4	20.01.2010	11:00:00	19205	18989	19205	18990	15398		20.01.2010	11:00:00	189,2
5	20.01.2010	12:00:00	18991	18905	19064	19037	25354		20.01.2010	12:00:00	188,58
6	20.01.2010	13:00:00	19034	18950	19109	18987	13272		20.01.2010	13:00:00	189,08
7	20.01.2010	14:00:00	18967	18932	19024	18997	8018		20.01.2010	14:00:00	188,6
8	20.01.2010	15:00:00	19000	18931	19010	18947	4808		20.01.2010	15:00:00	188,74
9	20.01.2010	16:00:00	18940	18888	18996	18989	15031		20.01.2010	16:00:00	188,11
10	20.01.2010	17:00:00	18966	18810	19013	18828	20665		20.01.2010	17:00:00	188,59
11	20.01.2010	18:00:00	18830	18785	18927	18876	17508		20.01.2010	18:00:00	187,1

› Рис. 4. Наша функция в состоянии разобрать такой лист с диапазонами данных.

Однако, если ваша функция ожидает видеть массив, а пользователь укажет не диапазон, а только одну ячейку, не ждите, что вы получите массив с одним элементом – на входе будет простая переменная. Перед вычислением такой функции следует проверить, что именно (массив или переменная) было ей передано:

```
If IsArray (aData) Then
Else
            'Код обработки ошибки
End If
```

Встроенная функция **IsArray()** возвращает **True**, если её параметр – массив. Более подробно о перехвате и обработке ошибок мы поговорим в следующий раз.

Наконец, некоторые функции могут даже возвращать массив. Они называются «функциями массива» (подробности ищите в справке по *OpenOffice.org*) и вводятся в ячейку с одновременным нажатием клавиш Ctrl+Shift+Enter, а результаты размещаются на заданной области рабочей книги. Пример простейшей функции массива есть на диске – это **Листинг 2** (рис. 3).

Чтобы отредактировать функцию массива, изменить значения её аргументов или удалить её с листа, нужно выделить занимаемую её результатом область вручную, либо отметить одну из ячеек диапазона и нажать Ctrl+/, где / – клавиша деления на цифровой клавиатуре. В данном примере результатом функции является массив чисел, но ничто не мешает возвращать массив фрагментов текста, или смешанный – из строк и чисел.

Читаем разметку

Можно подумать, что любая функция, кроме «Привет, мир!» и ей подобных, обязывает нас вводить данные в ячейки листа рабочей книги, а потом указывать ссылки на них или их диапазоны. Это правда, но не вся правда.

Можно, например, вводить нужные данные в заранее определенные ячейки на заранее определенные листы и считывать их в теле функции. Вполне допустимо получать данные из внешних текстовых файлов. Однако все эти варианты лишают функции универсальности, и применять данный подход целесообразно только в том случае, если вы пишете шаблон заранее определенного конкретного назначения. В некоторых случаях функцию удобно реализовать таким образом, чтобы она не брала данные из строго определенных мест, а разыскивала их по некоторым признакам самостоятельно. »

Рассмотрим для примера функцию, которая принимает два массива (диапазона), содержащих данные о торгах по двум финансовым инструментам, и вычисляет некоторый финансовый индикатор – в данном случае не важно, как и какой. Такой функции необходимо получать данные из двух диапазонов с непредсказуемым размером, так как при импорте он будет постоянно изменяться. Мы не будем каждый раз редактировать ячейку с вызовом функции, указывая новый диапазон – вместо этого будем разбирать разметку электронной таблицы. Самый простой способ реализации означенного механизма – использование на листе текстовых меток (ключевых слов), указывающих на начало таблицы. При этом функция должна ссылаться на заранее известный лист, содержащий диапазоны с данными. Фрагмент такого листа показан на рис. 4.

Здесь ячейки **A2** и **I2** содержат метки для первого и второго диапазона данных: **STOCK1** и **STOCK2**, соответственно. По правилам разметки для описываемой функции, метка диапазона находится над его крайней правой верхней ячейкой и диапазоны имеют постоянную ширину – то есть количество столбцов; а количество строк может изменятся в любых пределах. Кроме того, очевидно, что метки должны быть уникальны и встречаться на листе только один раз. Давайте рассмотрим работу функции, способной идентифицировать и загрузить данные из описанных диапазонов.

В первую очередь, сделаем лист с диапазонами непредопределённым – это значительно упростит использование функции. Для этого добавим входной параметр **nSheets**, в котором будем передавать имя нужного нам листа. Кроме того, дополним функцию ещё одним параметром, **ListenRange** – для чего он нужен, мы обсудим чуть ниже:

```
Public Function ArbitraryAnalysis(ByVal nSheets As String, ByVal
ListenRange As Variant) As Integer
```

В первую очередь необходимо на листе, имя которого передано в переменную **nSheets**, найти метки. Естественно, мы не будем перебирать поочерёдно все ячейки листа в цикле, а воспользуемся стандартным поисковым интерфейсом *OpenOffice.org* – **com.sun.star.util.XSearchable**:

```
oSheet = ThisComponent.Sheets.getByName(nSheets) ' лист
на который ссылается функция
oSearchDesc = oSheet.createSearchDescriptor() ' создаём
дескриптор поиска
oSearchDesc.SearchString = "STOCK1" ' настраиваем дескриптор
oFound = oSheet.findAll(oSearchDesc) ' получаем результат
поиска
oCellAdr0 = oFound(0).CellAddress ' получаем адрес первого
вхождения
```

Следующий этап – определение границ диапазона. Так как мы условились, что наши диапазоны не имеют пустых ячеек, тут всё достаточно просто – обнаружение первой же незаполненной ячейки ниже ячейки с меткой будет означать, что диапазон завершился.

› Рис. 5. Так выглядят массивы данных после анализа листа.

```
SCA = oCellAdr0.Column
SRA = oCellAdr0.Row + 1
ECA = SCA + 7 - 1 ' количество столбцов фиксированное,
нумерация с 0
ERA = SRA ' предполагаем что в диапазоне одна строка
Do While oSheet.getCellByPosition(SCA,ERA).getString <> "" '
пока ячейка не пуста...
            ERA=ERA+1 ' увеличиваем индекс
Loop
ERA=ERA-1 ' отступаем назад в диапазон
```

Пожалуйста, имейте в виду, что определение границ диапазона – задача достаточно длительная, и затрачиваемое на неё время напрямую зависит от размера диапазона.

Теперь, когда диапазон найден, извлечь требуемые данные в виде массива из него не представляет никакой сложности:

```
oTargetRange = oSheet.getCellRangeByPosition(SCA,SRA,ECA,
ERA) ' получаем диапазон с данными
stock1 = oTargetRange.getDataArray ' извлекаем данные
в массив
```

На рис. 5, демонстрирующем состояние массивов **stock1** и **stock2** в режиме отладки, видно, что оба диапазона данных успешно считаны и загружены в соответствующие переменные.

Поиск второго диапазона осуществляется аналогичным образом. Причём мы можем использовать тот же дескриптор поиска, заменив в нём только свойство **SearchString**: **oSearchDesc.SearchString = «STOCK2"**. Полный исходный текст функции можно найти на прилагаемом к журналу диске в файле **Листинг 3**.

Теперь самое время объяснить, для чего был добавлен второй аргумент функции – **ListenRange**. Если вы уже просмотрели полный листинг примера, то, вероятно, обратили внимание, что нигде в теле функции эта переменная не используется. Так для чего же она нужна?

Чтобы дать ответ на этот вопрос, необходимо пояснить, как *Calc* пересчитывает формулы. Это происходит лишь в том случае, если изменились какие-либо входящие в нее аргументы, и это вполне разумно – зачем тратить время на пересчёт функций, значения которых не изменится? Так как после загрузки диапазонов аргумент функции, принимающей лишь имя листа, не изменится, функция не будет пересчитана. Для целей запуска пересчёта мы и вводим ещё один аргумент – «прослушиваемый» диапазон. Им может быть любой диапазон, изменение в котором должно приводить к пересчёту функции; но наиболее интересным для нас будет указание диапазона, пересекающегося с диапазонами данных – например, **ЛистСДанными.A3:O6**, где **ЛистСДанными** (естественно) – имя листа с данными. Такая конструкция будет охватывать несколько строк одновременно из обоих загружаемых функцией диапазонов данных, и соответственно, её значение будет пересчитываться при всяком изменении в них.

Указание и поиск меток на листе не всегда приемлемы, так как это может приводить к коллизиям: например, если диапазоны данных могут содержать непредсказуемый текст, в том числе и совпадающий с меткой, или если диапазоны, по условию, могут иметь пустые ячейки, что затруднит поиск их границ. В этом случае правила разметки придётся усложнить. Можно ввести, например, проверку на стиль форматирования метки и просматривать найденные вхождения на предмет метки, имеющей заданный стиль, а диапазон разбирать не по одному, а по всем столбцам сразу. Но если уж мы используем стили, почему бы не отформатировать заданным стилем сам диапазон с данными?

Например, если в вышеописанном случае создать стили STOCK1 и STOCK2, для форматирования первого и второго диапазона с данными соответственно, то от использования текстовых меток можно отказаться совсем. Сама же функция при этом очень упростится, так что получать диапазон теперь можно будет

непосредственно как результат поиска, что одновременно снимает и проблему нахождения в нём пустых ячеек. Для поиска диапазона, отформатированного заданным стилем, достаточно установить его имя в качестве свойства **SearchString** дескриптора поиска, а свойство **SearchStyles** выставить в значение **True**, что активирует поиск по стилям:

```
oSearchDesc.SearchStyles = 1 ' устанавливаем флаг стилей
oSearchDesc.SearchString = "STOCK1" ' устанавливаем имя стиля
oFound = oSheet.findAll(oSearchDesc) ' получаем диапазон
с данными
oTargetRange = oFound(0)
stock1 = oTargetRange.getDataArray ' извлекаем данные в массив
```

Ещё одним преимуществом разметки стилями является значительный прирост производительности, так как больше нет необходимости искать границы диапазона данных в цикле. Это особенно ощутимо, если размер диапазона велик и непостоянен в обоих измерениях.

Конечно, можно придумать ещё более изощрённые и сложные методы разметки, но это уже частные вопросы реализации, которых мы касаться не будем.

Место для функции

Теперь, когда мы в принципе представляем, как выглядят пользовательские функции *OOo Calc*, у вас может возникнуть резонный вопрос: а где их вводить? Если обычные макросы могут размещаться в любых библиотеках Basic, то функции, особенно используемые в *Calc*, в этом вопросе несколько более прихотливы.

Обратите внимание на рис. 6. На нём представлено диалоговое окно Макрос из *OpenOffice.org Basic*. Если посмотреть повнимательнее, то можно заметить, что у библиотек, представленных в списке «Макрос из», используются собственные значки. Они показывают, загружена ли библиотека. На данном рисунке загружены библиотеки *Standard*, *OceanStarter* и *OceanRuntimeEnvironment*. Загрузка библиотеки происходит при первом обращении к ней, которое может происходить при запуске из неё макроса или просмотре её модулей.

Загрузку также можно выполнить и принудительно из кода на Basic:

```
BasicLibraries.LoadLibrary("MyLibrary")
```

Библиотека *Standard* загружается при старте *OpenOffice.org* автоматически.

Если вы откроете файл **.ods**, в котором используются функции Basic из незагруженной на данный момент библиотеки, то результатом всех таких функций будет ошибка **#ИМЯ**, так как *Calc* просто не сможет их найти. Более того, в ветке *OOo 3.x* функции *Calc* не будут работать в библиотеках, отличных от *Standard*, даже если такие библиотеки загружены. В этом случае все функции будут возвращать ошибку **#ЗНАЧЕН!**. Таким образом, наилучшим местом размещения пользовательских функций *Calc* в профиле пользователя будут модули библиотеки *Standard*.

Однако, если вам необходимо отправить файл *Calc*, использующий ваши функции, другому пользователю, и вы не уверены, есть ли у респондента необходимые модули Basic, или даже знаете, что их нет, тогда этот способ вам не подходит. В этом случае лучше воспользоваться внутренним контейнером библиотек самого файла. Все, что сказано выше, остается справедливым и в данном случае, за исключением того, что функции, размещенные вне библиотеки *Standard*, будут доступны и в *OOo 3.x* – после загрузки.

Передавая кому-либо файл, содержащий функции *Calc*, вы, вероятно, захотите защитить ваш код от случайного или даже преднамеренного изменения. Сделать это можно, просто открыв сохранённый файл **.ods** любым архиватором и, отыскав в **/Basic/Standard/script-lb.xml** строку

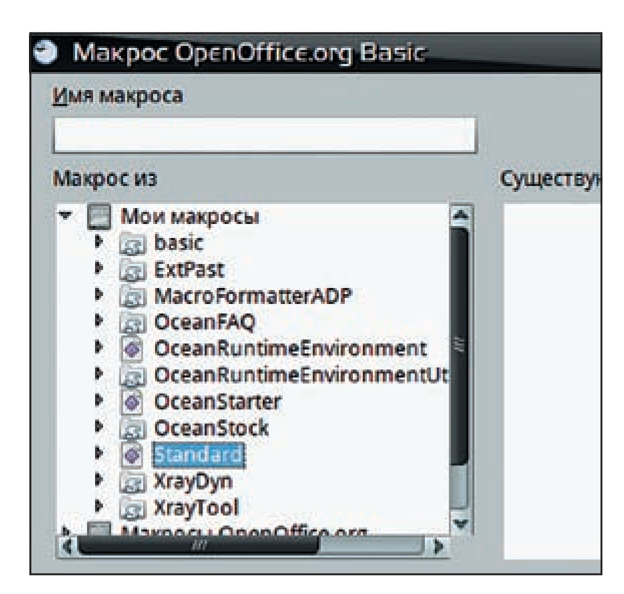
```
<library:library xmlns:library="http://openoffice.org/2000/library"
library:name="Standard" library:readonly="false" library:password
protected="false">
```

изменить в ней значение **library:readonly** с **false** на **true**. Это можно проделать и напрямую из Basic, установив для библиотеки флаг **ReadOnly**:

```
ThisComponent.BasicLibraries.setLibraryReadOnly("Standard", True)
```

Кроме того, нужно предостеречь пользователей, любящих размещать в одном модуле библиотеки сразу много функций. Помните: максимальный размер модуля библиотеки Basic составляет 65535 байт. Если вы превысите его, то ваш код будет оставаться работоспособным только до следующего перезапуска *OpenOffice.org*, так как при сохранении длина всех модулей будет урезана до 64K знаков.

Полагаю, вы уже поняли, что с помощью пользовательских функций *Calc* можно значительно расширить функционал электронных таблиц в самых различных направлениях. Этим-то мы и займемся через месяц. LXF



› Рис. 6. Диалоговое окно макросов и библиотеки Basic.

Немного про отладку

Читая эту статью, вы наверняка набросали для себя несколько примеров функций на *OpenOffice.org Basic* и теперь хотите убедиться в их работоспособности. Что же, и к этой задаче существует несколько подходов.

Один из наиболее удобных способов отладки функций, возвращающих численное значение – это построение диаграмм. Пользуясь графиком, удобно отслеживать зависимости между переменными и результатами, сообразуясь с их физическим смыслом.

Если вы воспроизводите в *Calc* функцию какого-либо внешнего пакета, логичным будет принять предположение о том, что в данном пакете функция реализована без ошибок. В этом случае можно воспользоваться заранее сгенерированным массивом её результатов, сопоставленных массиву аргументов, и сравнить его с тем, что выдает ваша функция при тех же самых входных величинах.

Не стоит и забывать о стандартных правилах проверки возвращаемых результатов. Для каждой функции очень желательно проверить результаты, выдаваемые для крайних значений из области определения функции и некоторых стандартных критичных значений. Например, если область определения вашей функции от -1 до 1, то обязательно проверьте значения, возвращаемые ею при -1 и 1, а также при 0. Ноль, как правило, всегда является критичным значениям. Другие неплохие кандидаты – -1 и 1, если, конечно, они не являются крайними для области определения, как в примере выше. Также критичными являются такие значения аргументов, при которых значение функции обращается в -1, 0 или 1 или же достигает своих экстремумов. По возможности, желательно проверить работу функции и с этими входящими параметрами.

» **Через месяц** Обработка ошибок, перегрузка функций и псевдонимы.

Python: Анима-

Clutter всецело ориентирован на разработку высококлассных пользовательских интерфейсов. **Ник Вейч** приправит графику, слегка вышколив кнопки.





Наш эксперт

Ник Вейч стремительно становится генералом ловли ляпов в модулях Python. К позорному столбу их!

Clutter был создан для «оживления» графических интерфейсов, и нам пора посмотреть, как из объектов-актеров создаются кнопки, а затем добавить к ним несколько занятных атрибутов. Но для начала, прежде чем демонстрировать поразительные новые навыки общения с мышью, надо припасти что-то, с чем мы будем взаимодействовать. Как и почти все инструментарии, *Clutter* является событийно-ориентированным. То есть, если в приложении происходит что-то значимое, генерируется сигнал. Грубо говоря, другие участки кода ждут этого сигнала и делают что-то в ответ.

Если в *Clutter* отметить актера как реагирующего [reactive], то при любом событии с ним – типа щелчка, переноса, движения над ним курсора мыши и даже набора на клавиатуре – он будет генерировать полный набор сигналов. Каждый из них можно подключить на уровне объекта к методу или функции обратного вызова, обеспечивающей реакцию на сигнал. Чтобы опробовать это, не потребуется даже создавать приложение: все делается в интерактивной оболочке Python. Откройте терминал и введите **python**, а затем начните ввод (если вы ленивы, можете скопировать код из файла-листинга, имеющегося на **LXF**DVD). Приступим:

```
import clutter

def entered(actor,event):
  print event,actor
  actor.set_color(clutter.Color(0,0,255,255))
  return '#t'

def exited(actor,event):
  print event,actor
  actor.set_color(clutter.Color(255,0,255,255))
  return '#t'
```

Здесь мы создали функции обратного вызова для двух событий. Неважно, что это за события, и даже неважно, какие объекты их генерируют – функции тождественны. На самом деле, при желании их можно объединить в одну, но наш вариант кода понятнее.

По сути, **entered()** и **exited()** – просто обычные функции. Обе получают два параметра, объекты **actor** [актер] и **event** [событие]. Вначале они выводят эту информацию (что удобно при отладке, для просмотра, какой объект что сгенерировал, но в итоговом приложении не нужно), затем используют унаследованный актером метод **set_color**, для изменения его цвета. Здесь мы просто определили объекты **clutter.Color** на лету. Они принимают значения RGBA, так вот мы установили синий в первом и фиолетовый во втором случае.

Отцы и дети

Последняя строка может вызвать удивление: зачем мы возвращаем значение из события? И почему такое странное? Происходит следующее: *Clutter* получает сигнал **event** на родительском объекте, сцене, которая всегда является реагирующей. В предыдущих уроках нашей серии мы просто соединяли сигналы напрямую от сцены – прошлый раз, например, в обработчике нажатий клавиш. Но родительский объект также просматривает всех своих потомков, пытаясь определить, кто из них ответственен за эффект. Это может быть, например, одинокий прямоугольник в углу или часть в большей группе объектов, являющейся потомком сцены. Когда «виновный» потомок найден, он генерирует сигнал, «всплывающий» вверх к родителю. Это необходимо потому, что если прямоугольник не имеет явного обработчика, его может

Что нам потребуется

Прежде чем начать, установите Python и его модуль *Clutter*. Если вы используете дистрибутив, обновленный не раньше, чем год назад, то оба доступны в ваших репозиториях. Обычно проще загрузить всё из них, но поискать исходные тексты последней версии *Clutter* на www.clutter-project.org. Чтобы добавить видео в конце урока, понадобится *GStreamer*, а также пакеты **cluttergst** и **pyclutter-gst**.

» **Месяц назад** Исследовали способы отрисовки необычных актеров в *Cairo*.

ция интерфейса

иметь родительский объект, и щелчок на прямоугольнике будет обрабатываться на уровне группы.

Этот процесс – перехват событий в так называемой «пузырьковой» фазе – действие стандартное, но оно тратит ресурсы, продолжая передавать событие по всем объектам, даже если оно уже обработано. Изящно пресечь это можно, прервав процесс, по типу выхода из цикла – именно это и происходит, когда функция обработки сигнала возвращает значение **#t**. И наоборот, если мы решаем, что еще не завершили обработку сигнала (или что-то сделали, но родительский объект все еще нуждается в уведомлении), можно вернуть **#f** для продолжения процесса. Таково поведение по умолчанию, но при написании кода почти всегда лучше все объявлять явно: впоследствии это спасает от изрядной головной боли.

Устраиваем сцену

Теперь, разобравшись с обработкой сигналов, придумаем несколько генерирующих их объектов, а также сцену, где мы всё разместим.

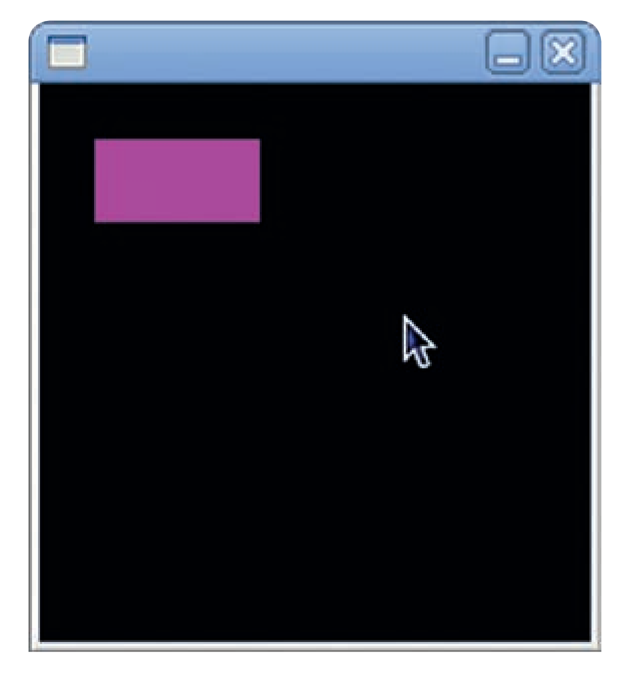
```
>>> stage=clutter.Stage()
>>> stage.set_color(clutter.Color(0,0,0,255))
>>> stage.set_size(200,200)
>>> r=clutter.Rectangle()
>>> r.set_size(60,30)
>>> r.set_color(clutter.Color(255,0,0,255))
>>> r.set_position(20,20)
>>> r.set_reactive(True)
```

Этот код должен быть уже знаком тем, кто следит за нашими уроками. Он создает сцену (главное окно) нашего скрипта и добавляет простой объект-актер – в данном случае, прямоугольник. Метод **set_reactive()** сообщает *Clutter*, что актеру нужно получать события и генерировать сигналы. Это подходит не всегда – фоновой текстуре, например, сигналы ни к чему, даже если она и перехватит пару щелчков – так что здесь требуется явный вызов.

Теперь необходимо просто соединить события с обработчиками, определенными нами ранее, добавить прямоугольник на сцену и показать все, отрисовав в окне:

```
>>> r.connect('enter-event', entered)
>>> r.connect('leave-event',exited)
>>> r.show()
>>> stage.add(r)
>>> stage.show_all()
```

Самые наблюдательные из вас могли уже спросить себя: как работает метод **connect()**? В конце концов, наши обработчики событий желают знать актера и событие, а все, что мы сделали – это соединили сигналы с соответствующим именем функции (пара скобок после entered или exited не нужна – это не вызов!). Так вот, метод **connect()** просто хранит соответствие имени события и функции, которую следует вызвать. Когда событие происходит, он вызывает функцию и автоматически передает ей два параметра: себя, (или, по крайней мере, указатель на родительский объект – в нашем случае, прямоугольник), а затем объект-



› Это не фиолетовый кирпич, а устройство сложного графического интерфейса пользователя.

событие. Последнее на самом деле содержит много информации, полезной для обработчика. Кроме типа события, это время (согласно часам *Clutter*), координаты x и y и сцена, где произошло событие, что удобно для многооконных приложений.

Попробуйте. Нам не нужно входить в основной цикл *Clutter*, чтобы заставить работать обработчики событий нашего скрипта: просто подвигайте мышью в окне над прямоугольником – и увидите, что цвет меняется. Отведите мышь, и он сменит цвет на другой. Ах, эта магия событий…

Возможно, мы пожелаем также перехватывать события щелчков, но вскоре увидим, что существует бездна сигналов, для которых следует создать функции обратного вызова, и все это только для одного объекта-актера!

»

Помехи на линии

Некоторые люди, в порядке, э-э, конструктивной обратной связи злопыхали, что в приложениях *Clutter* анимация выполняется не плавно, а иногда и вовсе не выполняется. Части таких проблем можно избежать, взяв другую версию библиотек *Clutter* (особенно если вы используете сборки до версии 1.0), но другая потенциальная проблема коренится не здесь.

Сервер *X.org X* не очень-то охотно работает с двумя экземплярами GL одновременно (что следовало бы учесть в будущем релизе), и если вы запускаете приложение *Clutter* вместе с чем-то, также использующим *GL/GLX*, то не исключены проблемы с производительностью. Сам я, если хочу, чтобы у моих окон колыхались края, просто пью сидр, но понимаю, что многие читатели норовят включить лихой *Compiz* ради визуальных эффектов. Почти наверняка это приведет к конфликтам с *Clutter*, так что хлебните чуть-чуть и отключите *Compiz*, если не хотите неприятностей в процессе урока.



› Четыре кнопки, созданные роботом! Ну ладно, циклом Python, по любому...

Решение заключается в том, чтобы ухитриться заставить наш обработчик работать с несколькими событиями. Кстати, он уже работает с несколькими актерами – посмотрите, ведь код ссылается на актера, подающего сигнал, а не на конкретный объект **r**. То есть, если мы создадим полторы дюжины прямоугольников, все они будут вести себя одинаково.

Давайте начнем заново, в этот раз с целой стаей кнопок, на которые можно нажимать, но с единственным обработчиком. Теперь несколько проще написать скрипт в *Kate* или *Gedit* и запустить его как обычное приложение:

```
>>> stage.remove(r)
>>> button =[]
>>> def handler(actor,event):
... print event,actor
... print event.type.value_nick
... if event.type.value_nick=='enter':
...    actor.set_color(clutter.Color(0,0,255,255))
... elif event.type.value_nick=='leave':
...    actor.set_color(clutter.Color(255,0,255,255))
... elif event.type.value_nick=='button-press':
...    actor.set_color(clutter.Color(255,255,255,255))
... return '#t'
...
>>> stage.set_size(300,200)
>>> for i in range(4):
... r=clutter.Rectangle()
... r.set_size(50,30)
... r.set_color(clutter.Color(255,0,0,255))
... r.set_position(25+(i*60),150)
... r.set_reactive(True)
... r.connect('enter-event', handler)
... r.connect('leave-event',handler)
... r.connect('button-press-event',handler)
... r.show()
... stage.add(r)
... button.append(r)
...
```

Здесь есть только два момента, достойных упоминания, поскольку они отличаются от того, что мы уже делали. Во-первых, обработчик теперь реагирует на все события от всех кнопок. Пока примем, что все кнопки ведут себя одинаково (к более изощренному сценарию мы перейдем позже), и все, что нам следует сделать – это сообразить, какое событие обрабатывает функция. Поскольку объект **event** автоматически передается в виде параметра, то необходимо просто рассмотреть его свойства (их много, но **event.name_nick** короткое и делает код более читабельным). В Python отсутствует конструкция 'case … switch' – однако от использования if и elif мы здесь потеряем не много.

Во-вторых, мы делаем нечто слегка ненормальное. В цикле **for** мы создаем прямоугольник **r** и наделяем его свойствами, а затем мы возвращаемся и делаем все по новой – а не затираем ли мы значение в **r**? И да, и нет. В самом конце цикла мы добавляем объект-прямоугольник к нашему списку кнопок методом **append()**. На самом деле, **r** – не сам объект, а лишь его адрес. Прямоугольник будет уничтожен только в том случае, если на него не останется действующих ссылок. В данном случае ссылка есть: она в нашем списке кнопок. При следующем проходе цикла создается еще один прямоугольник, и в **r** помещается новый адрес. Таким образом, каждая кнопка сохраняется, и все они различны. Другой способ сделать это – ссылаться на этапе создания на элементы в списке кнопок; но тогда следует заранее заполнить его корректным числом элементов или добавлять пустые объекты в начале цикла. Этот путь проще, хотя и заставляет делать двойную работу.

Хорошее поведение

Clutter нацелен на создание шикарных графических интерфейсов, так почему бы не сделать так, чтобы кнопки проявлялись и плавно исчезали при перемещении над ними мыши? Это пригодится для медиа-проигрывателя, в котором обычно вы смотрите на экран, но когда хотите чем-то управлять, появляются кнопки. Мы можем легко добавить их при помощи поведений [behaviour] *Clutter*. Поведение похоже на сохраненную анимацию, увязанную с событиями временной шкалы. Временная шкала, как мы видели в предыдущих учебниках, это просто механизм прерываний. *Clutter* позволяет подключать поведения к объектам, а затем вызывать анимацию, запуская шкалу времени.

Я знаю, о чем вы думаете: не проще ли было использовать метод **actor.animate()** для изменения прозрачности объекта? Про-ще-то проще, да только он не работает. В темных глубинах *Clutter* кроется разрушительная ошибка, не позволяющая использовать простой метод animate() для прозрачности. Если вы все же попытаетесь, то получите несколько предупреждений, хотя код все же будет работать, и любая анимация просто превратится в черноту. Поэтому поведение объектов – наше все.

```
>>> timelinefadein = clutter.Timeline(duration=600)
>>> timelinefadeout = clutter.Timeline(duration=600)
>>> alpha = clutter.Alpha(timelinefadein, clutter.EASE_IN_SINE)
```

Наскоро о Clutter

Если вы новичок в *Clutter*, вас, вероятно, выручат предыдущие выпуски этой серии (подпишитесь сейчас, чтобы получить бесплатный онлайн-доступ к PDF-файлам), но приведу краткую сводку.

Сцена *Clutter* – эквивалент объекта-окна на экране. Имеются различные методы для установки размера, цвета и подключения событий к сцене. Актер – это любой графический объект, появляющийся на сцене. Графические элементы, включая текстовые объекты и текстуры, порождаемые в *Clutter* или приходящие откуда-либо – это актеры. Они могут быть преобразованы в три измерения, и эти трансформации можно анимировать при помощи операций, обрабатываемых *Clutter*. Обычная процедура – настраиваем сцену, создаем актеров и добавляем их на сцену, а затем делаем что-нибудь крутое.

```
>>> behaviour = clutter.BehaviourOpacity(0x0, 0xc0, alpha)
>>> alpha2 = clutter.Alpha(timelinefadeout, clutter.EASE_OUT_SINE)
>>> behaviour2 = clutter.BehaviourOpacity(0xc0, 0, alpha2)
```

Мы можем подключать оба поведения к кнопкам по мере их создания и вызывать их из нашего обработчика событий. То есть проявляться и исчезать будут все кнопки, а не только та, над которой завис курсор мыши.

Объекты **BehaviourOpacity** принимают три параметра – **alpha**-объект (контролирующий значения анимации от кадра к кадру) и начальную и конечную непрозрачности. Здесь мы привели их в шестнадцатеричном формате; обычно это лучше, поскольку некоторые операции над альфа-эффектами весьма требовательны к получаемым значениям, а в таком варианте вы будете уверены, что оно преобразуется как 8-битное целое.

После применения поведения к объекту с помощью **behavior.apply(actor)**, оно может быть вызвано в любой момент, запуском шкалы времени. Единственная проблема с нашей анимацией сейчас заключается в том, что если очень быстро подносить и убирать курсор мыши, то анимации наложатся друг на друга, вызвав мерцание. Одним из способов исправления этого может быть использование невидимого прямоугольника, накрывающего всю группу кнопок, и применение поведения затухания с его помощью.

Итак, большой вопрос в том, какие чудеса мы извлечем из волшебной корзины Python, чтобы заставить каждую кнопку вести себя по-своему без добавления хрентиллиона строк кода в наш обработчик событий? А позвольте привлечь ваше внимание к экспонату А:

```
>>> example=clutter.Rectangle()
>>> example.set_size(60,20)
>>> example.MyMadeUpProperty = 8
>>> example.MyMadeUpProperty
8
>>>
```

Вот вам и магия Python. Без всякой мороки с изобретением новых классов и тому подобного, мы можем динамически добавлять свойства к существующим объектам. Итак, хотя актеры *Clutter* не имеют соответствующего места для хранения свойства 'при нажатии кнопки выполнить', мы просто можем добавить его позднее. Актер **rectangle** будет вести себя так же, как обычный прямоугольник, но мы можем приписать к нему любые невероятные выкрутасы, а то и полезные вещи.

Нажмите кнопку

Это приводит нас к некоторым дополнительным ценным свойствам Python. Почти все в мире Python – просто объекты, включая методы и функции. В сущности, метод – всего лишь объект со свойством **__call__**. Радоваться тут особо нечему, но зато можно выполнять следующее:

```
>>> dir
<built-in function dir>
>>> dir()
['__builtins__', '__doc__', '__name__', '__package__']
>>> x=dir
>>> x
<built-in function dir>
>>> x()
['__builtins__', '__doc__', '__name__', '__package__', 'x']
>>> x(x)
['__call__', '__class__', '__cmp__', '__delattr__', '__doc__', '__
eq__', '__format__', '__ge__',
'__getattribute__', '__gt__', '__ hash__', '__init__', '__le__', '__
lt__', '__module__',
'__name__', '__ne__', '__new__', '__reduce__', '__reduce_ex__',
'__repr__', '__self__',
'__setattr__', '__sizeof__', '__str__', '__ subclasshook__']
```



› О, восторг толпы при виде наших появляющихся-исчезающих кнопок, которые парят над видео!

Dir – встроенная функция Python, с которой мы уже встречались; в основном она используется для интроспекции и сообщает, что есть в объекте. Присваивая переменной **x** значение **dir** (заметьте, что мы не используем скобки в конце), мы создаем ссылку на эту функцию. В результате **x** будет вести себя как **dir**. Это действительно **dir**, просто с другой меткой. Теперь вы можете вызвать **x** так же, как вызывается **dir**.

Слияние двух этих фрагментов с нашим существующим кодом *Clutter* означает, что мы сможем определить методы для выполнения действий, а затем присвоить их новым свойствам, которые добавим к уже созданным объектам-кнопкам. Наш класс-обработчик все еще не требует выполнения чего-то особенного – при щелчке на кнопке он просто вызывает ее свойство-метод 'action' или как мы там пожелаем его назвать. Вам, возможно, кажется, что мы немного увлеклись идеей не писать дополнительный код; тут и правда экономится несколько строк, но на самом деле этот способ обработки объектов реализуется для лучшей читабельности и сопровождаемости участка кода. Функция **handler()** – лишь эффективная часть конструкции, или, скажем, коммутатор, соединяющий компоненты вместе при необходимости.

Применив это и позаимствовав код из приложения *GStreamer*, которое мы писали несколько выпусков назад, можно создать наш собственный простой медиа-плейер с кнопкой паузы, которая исчезает и появляется поверх видео, когда это требуется. Здесь нет места, чтобы привести весь листинг (большую часть его кода мы уже видели), но вы можете найти его (наряду с некоторыми другими) на LXFDVD. LXF

Чувак! А где же карта?

В прошлый раз я сдуру намекнул, что в этом номере мы рассмотрим прекрасное картографическое приложение *Clutter*. К сожалению, несмотря на то, что такое приложение появилось, оно недостаточно надежно работало со многими версиями зависимых библиотек. Через несколько месяцев мы попробуем вернуться к этому, так что следите за событиями.

» **Через месяц** Сосредоточимся на *Cogls*, суб-атомной магии, стоящей за *Clutter*.

Google Cal: SMS

Клаудио Танчони опишет, как превратить Google Calendar в службу SMS-уведомлений, не обращаясь к платным сервисам.





Наш эксперт

Клаудио Танчони работает системным администратором, а Linux он полюбил с первого же взгляда.

Владельцы учетной записи на Gmail также получают доступ к другим инструментам и сервисам Google: Документам, Reader'у и, конечно, Календарю. На данном уроке мы покажем, как превратить последний в SMS-шлюз, посылающий вам SMS при возникновении некого события. Для примера мы настроим простенькую систему (состоящую из web-страницы на PHP, карманной базы MySQL и планировщика *Cron*), предназначенную для мониторинга сейсмической активности и посылающую нам сообщение при каждом подземном толчке.

Календарь Google не отличается от других похожих продуктов – кроме, конечно, его бесплатности для некоммерческого применения. Посылаемые сообщения также не будут стоить ничего, если ваш тариф не предусматривает платы за входящие SMS (если вы, скажем, находитесь в роуминге, плата может и взиматься).

Первым делом зарегистрируйте свой телефонный номер во вкладке Настройка для мобильных устройств на странице Настройки календаря. Убедитесь, что ваш мобильный оператор поддерживается, затем выберите страну, введите номер телефона и щелкните по Отправить проверочный код. Вам придет проверочное SMS с кодом, который нужно ввести в поле Проверочный код. Если все прошло нормально, вы увидите сообщение, что ваш номер телефона успешно верифицирован.

Далее, создадим мероприятие для тестирования системы. Зайдем в основной вид Календаря и щелкнем на колонке с сегодняшней датой пониже красной линии, отображающей текущее время. Введите название мероприятия в появившемся окне и щелкните на Изменить информацию для доступа к странице, где вводятся подробности о мероприятии и создаются напоминания (в области Напоминание). На каждое мероприятие можно назначать до пяти напоминаний.

Используйте первый выпадающий список для указания способа получения напоминания, а поле и список правее позволят сообщить, когда вы хотите получить его. Для целей тестирования мы выберем отправку SMS через **0** минут. Другими словами, вы получите сообщение точно в момент начала встречи. Если вы почему-либо не видите выпадающих меню в области напоминаний, щелкните по Добавить напоминание.

Закончив, нажмите Сохранить и ждите наступления события, следя за вашим сотовым телефоном. Если SMS придет, то ваш телефон работает с Календарем Google! Теперь пойдем дальше и погрузимся чуть глубже в особенности текстовых сообщений.

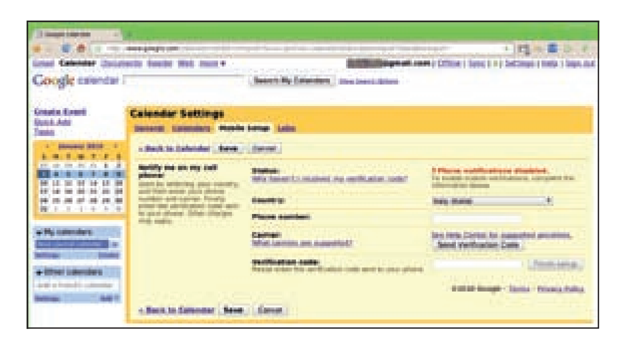
ПроZendируем тему

Разобравшись, как добавляются текстовые сообщения в браузере, давайте сделаем это, используя код на PHP и *Zend Framework*.

Zend (http://framework.zend.com) – это коллекция библиотек, предлагающая хорошо документированный набор классов и методов для работы с Календарем Google. Если у вас есть учетная запись Google, зарегистрированный в Календаре Google телефон и минимальная установка *Zend Framework* на вашем сервере web-приложений, ничто не мешает запустить данную примочку.

Наша задача теперь – отправлять текстовые сообщения всегда, когда выполняется заданное условие. И это легко сделать с помощью PHP-библиотек *Zend*.

Тут вы можете начать недоумевать, зачем вам мучаться с Календарем Google при наличии прекрасных доступных сервисов



› **На странице Настройка для мобильных устройств Календаря Google проверьте, поддерживает ли текстовые оповещения ваш мобильный оператор.**

» **Месяц назад** Применяли RAID, чтобы спасти ваши данные при сбое диска.

безвозмездно

SMS-шлюзов. Основная причина в том, что эти шлюзы не бесплатны; и хотя возможности безвозмездной посылки текстовых сообщений из Календаря Google несколько ограничены, с небольшими доделками это будет отлично работать на вас.

Для следующего примера займемся web-страницей, динамически обновляемой новым содержимым. А мы хотим, чтобы при появлении нового содержимого оно подчищалось, урезалось и посылалось нам на телефон, с помощью вскрытых возможностей отправки SMS в Календаре Google.

Используемая нами HTML-страница, обновляемая каждые несколько минут, содержит таблицу с землетрясениями в обратном хронологическом порядке, чтобы последние события были вверху. Данные приходят с официальной web-страницы, обновляемой геологическим сейсмическим институтом на http://bit.ly/daOfuO. Для нашего примера разместим таблицу с этой страницы на http://gcal-smsdemo.co.cc/earthquakes.php. При новом проявлении сейсмической активности строка об этом добавится в начале HTML-таблицы.

Первым шагом применим для чтения web-страницы функцию PHP **curl_exec**, затем выполним синтаксический разбор HTML-кода для проверки, не зарегистрировано ли новое землетрясение в начале таблицы.

```
/* Забираем html-документ для разбора */
$ch = curl_init("http://gcal-smsdemo.co.cc/earthquakes.php");
curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);
$html = curl_exec($ch);
curl_close($ch);
```

Функция **curl_init()** инициализирует сессию **Curl** и принимает URL как параметр. **Curl_setopt()** задает параметры для **curl_exec()**. **CURLOPT_RETURNTRANSFER, TRUE** велит следующей функции, **curl_exec()**, вернуть результат как строку (вместо булевского значения по умолчанию).

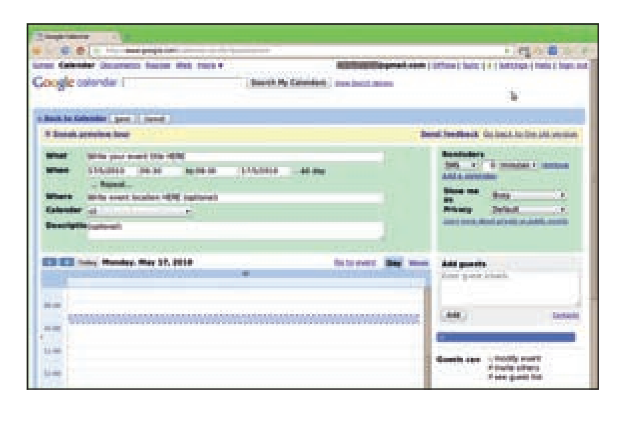
Далее, извлечем строки из таблицы с использованием DOM:

```
/* Новый объект dom */
$dom = new domDocument;
/* Загружаем html в объект */
$dom->loadHTML($html);
/* Массив таблиц по тэгу */
$tables = $dom->getElementsByTagName('table');
/* Забираем все строки из первой таблицы на странице */
$rows=$tables->item(0)-> getElementsByTagName('tr');
```

После загрузки HTML-страницы в объект DOM удобный метод **getElementsByTagName()** загружает все таблицы в массив **$tables** (на нашей демо-странице она только одна). Используя метод во второй раз, мы сохраняем все строки таблицы, которые вернул **getElementsByTagName('tr')**, в массиве **$rows**. Помните, что **item(0)** означает первую таблицу, так как индексы отсчитываются с **0**.

Пускай теперь на web-страницу только что добавилась строка с **Id 8211226050**, и нам нужно забрать ее, почистить, сжать и использовать наш трюк с Календарем Google для отправки SMS-сообщения. Давайте посмотрим, как это сделать.

Интересующая нас строка таблицы расположена на втором месте (в массиве **$rows** это элемент с номером один). Здесь важно отметить, что нам доступны только 60 символов SMS вместо обычных 160 [в русской кодировке вообще 74, – *прим. пер.*]. Так как не все колонки таблицы имеют одинаковую ценность, отберем только часть из них, чтобы уложиться в этот лимит.



›В Календаре Google можно задать, когда посылать вам сообщение о событии.

Интересные моменты

Интереснее всего будут **Seismic District** [Район землетрясения], **Date** [Дата], **Time(UTC)** [Время] и **Mag**(nitude) [Сила]. Параметры **Depth** [Глубина], **Lat**(itude) [Широта] и **Lon**(gitude) [Долгота] – более технические, и мы попытаемся немного поманипулировать строками, чтобы их тоже втиснуть, отказавшись от **Id**. Следующий код просмотрит массив **$rows** и создаст строку **$smstxt** с посылаемым текстом.

```
/* Выбираем каждую колонку по тэгу */
$cols=$rows->item(1)->getElementsByTagName('td');
/* создаем строку $smstext */
/* Location(9) + Magnitude(7) + TimeDate (3)(2) +
   Latitude (4) + Longitude(5) + Depth (6) */
$smstext=substr($cols->item(9)->nodeValue,0,8) .
   " " . $cols->item(7)->nodeValue . " " .
   $cols->item(3)->nodeValue . " " .
   $cols->item(2)->nodeValue . " " .
   $cols->item(4)->nodeValue . " " .
   $cols->item(5)->nodeValue . " " .
   $cols->item(6)->nodeValue . "Km";
```

Массив **$cols** содержит 10 элементов (0–9) с данными о последнем сейсмическом событии. Используя нужные индексы и функцию **substr()**, мы легко создаем SMS-сообщение.

После отработки кода над предыдущей таблицей, **$smstext** будет содержать нечто вроде **Chile-Ar Mb:6.2 11:25:38 2010/02/28 -34.739 -70.866 35Km**. Отметим, что это меньше, чем 60 символов, частично благодаря **substr($cols->item(9)->nodeValue,0,8)**, которая урезает поле Seismic District до восьми символов. Строка укорочена, и наше текстовое сообщение почти готово к отправке.

На следующем шаге применим логику, чтобы сообщение о сейсмическом событии посылалось однократно. Календарь Google не в состоянии определить, было ли отправлено SMS. Поэтому заведем небольшую базу *MySQL* для отслеживания уже посланных сообщений. Это может показаться лишним услож- »

Скорая помощь

Календарь Google записывает в текстовое сообщение только название события, длиной не более 60 символов. Остальное место используется для имени календаря, даты и времени начала/окончания события.

нением на пути к цели, но зато, когда база данных будет настроена, проверка, были ли сообщение уже отправлено, будет очень простой, потому что база не принимает одну и ту же запись более одного раза. А значит, в Календарь Google событие добавится только один раз, и он не будет зря бомбить наш телефон «эсэмэсками».

База данных в нашем примере будет содержать одну таблицу с именем **sms**, о двух колонках: **smstext** и **status**.

```
CREATE TABLE IF NOT EXISTS `sms` (
 `smstext` varchar(160) NOT NULL,
 `status` varchar(32) NOT NULL,
 PRIMARY KEY (`smstext`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

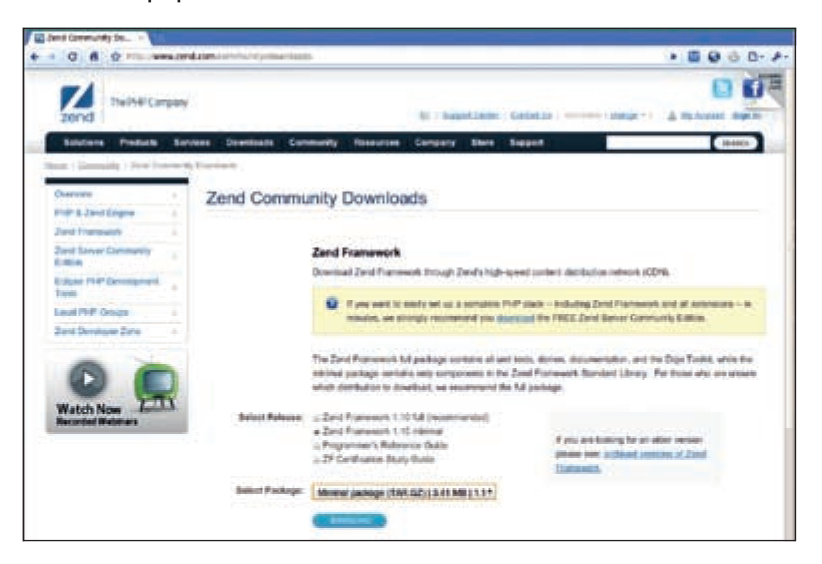
База сообщений

Поле **status** поможет нам отслеживать, было ли послано SMS, а **smstext** будет хранить его текст. Давайте вернемся к PHP-коду. Он вставляет текстовое сообщение в таблицу **sms** базы данных и сигнализирует об ошибке, если такое уже существует.

```
/** Соединяемся с БД **/
$con = mysql_connect('localhost', 'username','pa$$w0rd');
if (!$con) {
die('Could not connect: ' . mysql_error());
}
mysql_select_db("smsdemo_DB", $con);
/* Обезопасимся от инъекций и спец. символов */
$smstext = mysql_real_escape_string($smstext);
/** DB sms INSERT **/
$sql_ins_sms="INSERT INTO sms (smstext, status)
VALUES
('$smstext','pending')";
if (!mysql_query($sql_ins_sms,$con)) {
 print("Warning: " . mysql_error() . "<br />\n");
} else print("1 sms added<br />\n");
```

Метод **mysql_real_escape_string()** предупреждает вторжения в базу данных. Его следует использовать всегда, когда переменная содержит сведения из Интернета (разбора HTML) или введенные непосредственно пользователем (через формы). Оператор **INSERT** записывает в таблицу сообщение (**$smstext**), а также статус **pending**, означающий постановку сообщения в очередь на отправку. Следующим шагом будет запрос базы данных на наличие сообщений в очереди, и если таковые имеются – их отсылка. Этот код не отслеживает возможные ошибки, но вы можете позже добавить конструкции **try/catch** для обслуживания проблем с соединением и тому подобного. Сейчас же мы постараемся придерживаться краткого и лаконичного кода.

› Прежде чем пробовать это, загрузите и установите *Zend* на ваш сервер.



Ну вот, мы готовы использовать PHP для общения с Календарем Google. Для этого надо скачать и установить в поддиректорию минимальный *Zend Framework*. Также будем предполагать, что библиотеки *Zend* расположены в папке с именем **Zend**. Сперва загрузим некоторые классы *Zend*:

```
// Gcal Hack: Загружаем нужные классы Zends
require_once 'Zend/Loader.php';
Zend_Loader::loadClass('Zend_Gdata');
Zend_Loader::loadClass('Zend_Gdata_ClientLogin');
Zend_Loader::loadClass('Zend_Gdata_Calendar');
Zend_Loader::loadClass('Zend_Http_Client');
```

Эти классы PHP хранятся в папке **Zend_Gdata**; они позволят нам выполнять аутенфикацию, создавать новые мероприятия и удалять существующие, просматривать их, а также многое другое. Для начала войдем в нашу учетную запись Календаря Google с помощью такого кода:

```
// Подключаемся к сервису
$gcal = Zend_Gdata_Calendar::AUTH_SERVICE_NAME;
$user = 'username'; // без '@gmail.com' на конце
$pass = 'pa$$w0rd';
$client = Zend_Gdata_ClientLogin::getHttpClient ($user,
          $pass, $gcal);
```

$client будет хранить объект **Zend_Http_Client** со всеми деталями текущей аутенфикации клиента.

Далее проверим, нет ли сообщений со статусом **pending** в нашей таблице **sms**, и получим их, используя **SELECT**. Если в очереди есть какие-либо сообщения, первое из них будет выбрано для отправки:

```
// Создаем объект gcal
$gcal = new Zend_Gdata_Calendar($client);
// Получаем ожидающие события из таблицы sms
$sql_sel_pending = "SELECT * FROM sms WHERE
          status='pending' LIMIT 0,1";
$result = mysql_query($sql_sel_pending);
$row = mysql_fetch_array($result);
if (!$row) die("No pending event, exiting..." .
          "<br>\n");
// Устанавливаем заголовок
$title = $row['smstext'];
print("title: " . $title . "<br />\n");
```

В этом коде **$gcal** – объект **Zend_Gdata_Calendar**, представляющий событие нашего календаря с полем **title** (название), временем начала/конца и всем прочим, что используется при планировании мероприятия. Порция кода PHP/*MySQL* получает текстовые сообщения из очереди оператором **SELECT** и обрабатывает результат с помощью функции **mysql_fetch_array()**, которая возвращает массив с соответствующими строками, а **$title** в результате будет хранить текст, который надо послать.

Играем со временем

Календарь Google обычно подразумевает установку напоминаний о событиях, которые произойдут в будущем. Чтобы обойти это и обеспечить отправку сообщений на ваш телефон в реальном времени, нужно взять текущее время и добавить к нему небольшой сдвиг, чтобы Google считал ваше событие фактом ближайшего будущего. Пара минут (120 секунд) будет в самый раз. Это сработает, даже если вы установите одну минуту, но тогда сообщения могут иногда теряться при задержках в линиях связи.

```
// Задаем время
date_default_timezone_set('UTC');
```

» **Пропустили номер?** Узнайте на с. 107, как получить его прямо сейчас.

```
$safeTimeToWait = 120; // добавить секунд
$now = time(); // секунд с 1/1/1970
$smsSchedTime = $now + $safeTimeToWait;
print ("sms scheduled sending time is: " .
date("Y-m-d H:i",$smsSchedTime) . "<br />\n");
$sdate_yy = $edate_yy = date("Y",$smsSchedTime);
$sdate_mm = $edate_mm = date("m",$smsSchedTime);
$sdate_dd = $edate_dd = date("d",$smsSchedTime);
$sdate_hh = $edate_hh = date("H",$smsSchedTime);
$sdate_ii = $edate_ii = date("i",$smsSchedTime);
$start = $end = date(DATE_ATOM, mktime($sdate_hh,
        $sdate_ii,
        0,
        $sdate_mm,
        $sdate_dd,
        $sdate_yy));
```

Самая важная переменная тут – это **$smsSchedTime**: она задает время отправки текстового сообщения. Она получается сложением текущего времени (возвращаемого функцией **time()**) и **$safeTimeToWait**.

Готовность к отправке

Оператор **print** затем покажет результирующее время отправки. Переменная **$sdate_** содержит текущий год (**yy**), месяц (**mm**), день (**dd**), час (**hh**) и минуты (**ii**). Мы должны задать время начала равным времени окончания, потому что это не реальное мероприятие. Далее мы задаем событие как происходящее через две минуты в будущем и включаем текстовое напоминание:

```
// Конструируем объект-событие
$event = $gcal->newEventEntry();
$event->title = $gcal->newTitle($title);
$when = $gcal->newWhen();
$when->startTime = $start;
$when->endTime = $end;
$event->when = array($when);
// Настраиваем и включаем SMS-напоминание
$minutes = 0;
$method = "sms";
$reminder = $gcal->newReminder();
$reminder->method = $method;
$reminder->minutes = $minutes;
$when = $event->when[0];
$when->reminders = array($reminder);
$gcal->insertEvent($event);
print("Gcal Event Added, user: $user<br />\n");
```

Здесь создается объект **$event**, устанавливается **title** (тело сообщения), затем загружаются времена **startTime** и **endTime** (эти две переменные одинаковы и хранят время отправки нашего «сейсмического» SMS). Вас не должно смущать **$minutes = 0**. Это означает, что мы не хотим получать напоминания от Google заранее. **$method = «sms»** задает способ, которым мы хотим быть проинформированы, а затем событие **gcal** вставляется в календарь при помощи **$gcal->insertEvent($event)**.

Наконец, изменим статус нашего сообщения с **pending** на **sent** в таблице **sms**:

```
// Изменяем статус 'field' с pending -на -> sent
$sql_update_status = "UPDATE sms
        SET status = 'sent'
        WHERE smstext = '$title'";
if (!mysql_query($sql_update_status,$con)) {
print('Error: ' . mysql_error());
} else {
print("sms: $title [status=sent]<br />\n");
}
/** закрываем соединение с БД **/
mysql_close($con);
```

Что нам надо: LAMP + Zend

Linux *Apache MySQL* PHP (LAMP) – это наиболее популярная серверная платформа. Установить сервер LAMP на вашем компьютере относительно просто (см. по ссылке ru.wikipedia.org/wiki/LAMP). Мы настоятельно рекомендуем сперва попробовать ее бесплатно в Интернете. Поищите слова «бесплатный хостинг LAMP».

Zend Framework – открытый объектно-ориентированный каркас приложений для PHP5. Для его использования на вашем LAMP-сервере скачайте минимальный пакет *Zend*, распакуйте его и загрузите папку **/ZendFramework-1.9.7-minimal/library/Zend** куда-нибудь в директорию **/public_html** на вашем web-сервере.

Страница PHP теперь готова и может быть просмотрена онлайн на http://gcal-smsdemo.co.cc/smsdemo.php. Нам нужен кто-то или что-то, вызывающее созданную страницу. К счастью, *Cron* для этого идеально подходит! Настроим задание *Cron* на выполнение каждые 10 минут или около того (так как известно, что страница сейсмического института обновляется каждые несколько минут), и можно еще перенаправить стандартный вывод и стандартные ошибки в файл журнала. Запись crontab может выглядеть примерно так:

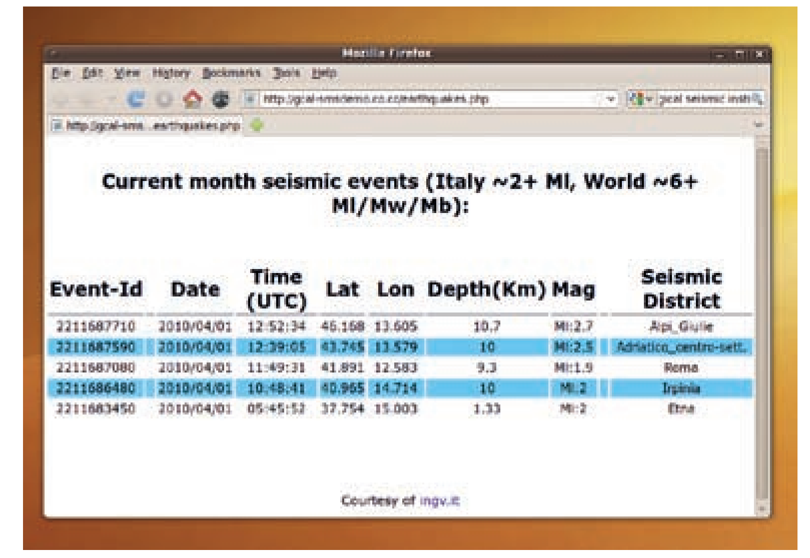
```
0,10,20,30,40,50 * * * * curl --silent --show-error --connect-
timeout 30 --max-time 40 http://gcal-smsdemo.co.cc/
smsdemo.php >> curl.log 2>&1
```

Параметры **silent** и **show-error** сделают вывод команды **curl** лаконичным, но более значимым в случае ошибок. **Connect-timeout** и **max-time** нужны на случай обычной опасности временного сбоя DNS, который может подвесить команду на неопределенный срок. Поэтому здесь, если что-то пойдет неправильно, мы просто прикроем команду через 30 секунд, если не будет ответа от web-страницы (упал сервер) или вся операция (полное время запуска PHP-страницы) занимает слишком долгое время (более 40 секунд).

Итак, мы сделали это. Теперь вы сможете превратить вашу учетную запись Календаря Google в шлюз текстовых сообщений, который будет эффективно посылать вам SMS в режиме реального времени. LXF



Календарь Goggle позволяет назначать мероприятия на прошлое, но тогда вы не получите текстовые напоминания.



› Наш сайт сейсмических данных не особо элегантен, но этого ему и не нужно — он всего лишь ступенька к получению сведений на наш телефон.

» **Через месяц** Как контроль версий спасает всю вашу цифровую жизнь.



Ответы

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru

В этом месяце мы ответим на вопросы про:

1. Защиту от вирусов
2. Поиск числовых UID
3. Отдельный домашний каталог
4. Аппаратные сбои
5. Экзамены LPI
6. Случайные номера портов
7. Автомонтирование сервера
8. Исправление ошибки Grub в системе с двойной загрузкой
9. Поддержку AAC в Amarok
10. Установку Skype в Mandriva
11. Синхронизацию каталогов

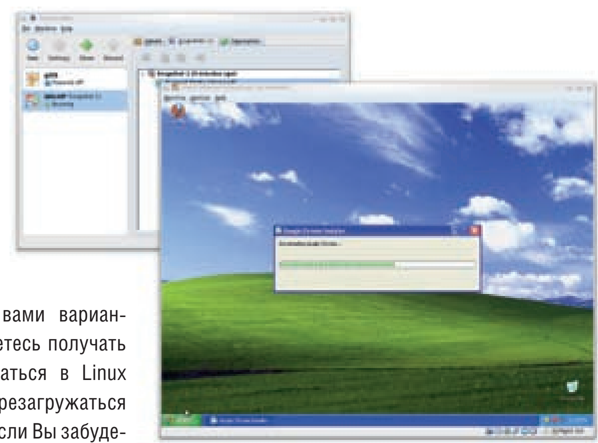
1 Защита Windows

В Я часто посещаю страну, где для компьютерных вирусов дом родной, и по работе должен принимать файлы от местных специалистов, нередко — зараженные. По требованию моего работодателя, я вожу с собой ноутбук с Windows XP, где поставлено регулярно обновляемое антивирусное ПО. И все-таки мой компьютер не раз инфицировался. Теперь во время приема файлов я хочу запускать Linux (в системе с двойной загрузкой, с помощью виртуализации, с носителя флэш-USB, или каким-либо другим способом). Поможет ли мне это избежать заражения?

Джон Соркин [John Sorkin]

О Антивирусные сканеры Linux умеют обнаруживать вирусы Windows – фактически, это их основная функция. Сравнительный анализ антивирусных программ публиковался в **LXF128** в рубрике Сравнение, и в этой статье Вы найдете больше информации, чем поместилось бы здесь. По-моему, предложенные вами варианты чреваты проблемой. Вы собираетесь получать файлы в Windows, затем загружаться в Linux для их проверки, а затем снова перезагружаться в Windows и пользоваться ими. Но если Вы забудете о перезагрузке/проверке/второй перезагрузке, достаточно открыть один плохой файл, и Ваш компьютер опять заразится. Запускать Linux из виртуальной машины можно, но тогда, чтобы сканер мог обнаружить вирус, Вам потребуется предоставить в общий доступ через *Samba* весь Ваш Windows-раздел. А при этом возникнут проблемы уже с безопасностью.

Поскольку для Вашей работы необходима Windows, я предложил бы запускать в виртуальной машине, наоборот, Windows, а Linux использовать как хост-систему. В этом случае Вы будете получать файлы, работая в Linux с активным антивирусным сканером, и передавать их в виртуальную машину Windows только после проверки. Кроме того, Вы будете иметь дополнительную страховку на тот случай, если инфицированный файл все же просочится. Программы виртуализации наподобие *VMWare Workstation* и *VirtualBox* умеют создавать «моментальные снимки» [snapshots] текущего состояния виртуальной машины. Если делать это регулярно, то в случае проникновения вируса (а в реальности такие шансы есть) для преодоления последствий заражения достаточно будет просто откатить виртуальную машину к предыдущему снимку. При этом Вы потеряете файлы, сохраненные после создания последнего «моментального снимка», но если для получения файлов, чтения электронной почты и выполнения других повседневных задач Вы пользуетесь дистрибутивом Linux, Ваши труды пострадать не должны.



› Запускайте Windows в виртуальной машине, а частые «снимки» защитят вас от вирусов.

В *VirtualBox* Вам придется создавать снимки вручную – но для этого достаточно просто не забывать регулярно нажимать кнопку, выполняющую данную операцию. *VMWare Workstation* может делать «моментальный снимок» после каждой остановки виртуальной машины, создавая резервные копии ежедневно или даже чаще. «Моментальные снимки» копируют только те данные, которые изменились с момента создания предыдущей копии, и не особенно поглощают дисковое пространство. **НБ**

2 Неконтактная Fedora

В Я пенсионер, и на досуге довольно много экспериментирую с разными дистрибутивами Linux. Недавно я установил Puppy Linux, Ubuntu 9.10, Linux Mint 8 и Debian. А вот Fedora 12 только что удалил.



Наши эксперты

» Мы найдем ответы на любой вопрос — от проблем с установкой системы или модемом до сетевого администрирования; главное — спросить!



Нейл Ботвик
Владелец ISP и экс-редактор дисков для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.

Майк Сондерс
Майк был одним из создателей прототипа **LXF** – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.

Пол Хадсон
Пол – местный супер-программист, и он может и хочет управиться со всеми вашими проблемами по части web и баз данных.



Грэм Моррисон
Когда Грэм не обозревает кучи программного обеспечения и не халтурит с MythTV, он готов к ответам насчет любого оборудования и проблем виртуализации.



Евгений Крестников
10 лет работал системным администратором. Он поможет вам разобраться с нетривиальными вопросами настройки Linux, а также со встраиваемыми системами.



Александр Казанцев
Разработчик Mandriva с радостью поможет вам решить проблемы с этим дистрибутивом.

КУДА ПОСЫЛАТЬ ВОПРОСЫ
Пишите нам по адресу: answers@linuxformat.ru **или спрашивайте на форуме:** www.linuxformat.ru

Чтобы получать доступ и обновлять мои файлы с любого из используемых дистрибутивов, я создал для них специальный каталог на отдельном разделе. Когда я первый раз начинаю пользоваться любым из дистрибутивов, я создаю директорию /home/graham/Mydata **и помещаю в** fstab **следующую запись:**

```
/dev/sda1 /home/graham/Mydata ext3 user,exec 0 0
```

затем вручную монтирую /dev/sda1 **и запускаю команду:**

```
chown -R graham /home/graham/Mydata
```

При каждой загрузке любого из дистрибутивов раздел с данными монтируется автоматически, и все файлы, в том числе и почтовые сообщения *Thunderbird*, для которых я отредактировал profiles.ini**, оказываются доступными, в том числе и для изменения.**

Эта система работает хорошо со всеми дистрибутивами, которые бы я ни попробовал — кроме Fedora. При запуске Fedora, стоит мне смонтировать свой раздел данных, Fedora меняет владельца моих файлов или на 500, или на 1000 (я и знать не знаю, что это за 500 или 1000).

В результате, когда я пытаюсь получить доступ к файлам данных из другого дистрибутива, оказывается, что у меня нет нужных прав. Мне приходится изменять их вручную. Я попытался блокировать SELinux в Fedora, и это смягчило проблему, но не ликвидировало ее полностью.

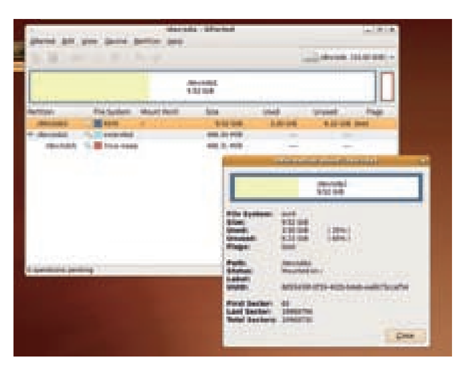
Грэм Холмс [Graham Holmes]

О Числа 500 и 1000 – это идентификаторы пользователя [user ID, UID]. Каждый пользователь, кроме имени [username], имеет еще и числовой идентификатор. Файловые системы хранят информацию о владельцах, используя эти числа, а система преобразует числовые идентификаторы в имена на основе информации из файла **/etc/passwd**. UID, меньшие некого значения, зарезервированы для системных пользователей – например, серверов и других демонов; поэтому пользователи имеют UID, начинающиеся с 500 или 1000, в зависимости от настроек дистрибутива. Ваш одиночный новый пользователь получает первый доступный номер. Для Fedora UID начинаются с 500, а для остальных упомянутых вами дистрибутивов – с 1000. Жесткого стандарта в данном вопросе нет. 500 было популярной начальной точкой, но большинство дистрибутивов теперь используют 1000.

При каждом запуске команды **chown** Вы переключаете UID владельца с 500 на 1000 и обратно, что не есть хорошо. Чтобы это пресечь, замените числовой UID Вашего пользователя в Fedora, чтобы он соответствовал UID во всех остальных системах. Это делается командой **usermod**. От имени root введите следующие команды:

```
usermod -uid 1000 graham
chown -R graham: ~graham
```

Первая команда изменяет Ваш UID на 1000, а вторая устанавливает владельца для всего содержимого Вашего домашнего каталога, включая Ваш раздел с данными, так, чтобы они совпадали. Команда **chown** не является абсолютной необходимостью: права доступа изменила бы для



› *GParted* **может изменить размеры корневого раздела, высвободив место для нового /home.**

Вас и **usermod**. Но лучше перебдеть, чем недобдеть. Кстати, информацию о том, какой UID Вам выделен, можно увидеть в файле **/etc/passwd**, по команде

```
grep graham /etc/passwd
```

Числовые идентификаторы пользователей и групп содержатся в третьем и четвертом полях. Более простой вариант – через команду **id**. Запуск ее без аргументов отобразит информацию о текущем пользователе. Если в качестве параметра указать имя пользователя, команда выведет подробные сведения о нем. **MC**

3 Дом переезжает

В LXF128, в разделе «Часто задаваемые вопросы» этой рубрики, упоминалось создание отдельного домашнего каталога и преимущества, предоставляемые этим решением. Я попытался поискать подробные инструкции о том, как это делается, на форумах LXF и в Google, но безуспешно. Результаты, выдаваемые Google, устарели, и часто сопровождаются предупреждениями.

Я очень хочу создать такой каталог /home заранее, перед выходом следующей LTS-версии Ubuntu. Пока я использую Karmic, и хотел бы на первых порах иметь возможность работать как с Karmic, так и с Lucid, причем без необходимости постоянно копировать данные туда и обратно.

Frustrated, с форумов

О Для Вашей задачи имеется два решения, в зависимости от того, есть ли у Вас дополнительный носитель, куда можно скопировать содержимое Вашего домашнего каталога (например, внешний жесткий диск). Это более простой вариант – в противном случае Вам придется «перетасовать» содержимое Вашего жесткого диска. В обоих случаях понадобится загрузиться с Live CD – лучше всего Live-версии Вашего текущего дистрибутива: так Вам будет проще ориентироваться, и она у Вас, наверное, уже есть.

Ваш внешний жесткий диск нужно отформатировать под одну из файловых систем Linux, в противном случае Вы потеряете метаданные – например, информацию о владельцах файлов и правах доступа. Когда винчестер будет готов, скопируйте содержимое каталога **/home**, используя свой любимый файловый менеджер. Обратите внимание, что мы копируем весь раздел **/home**, а не просто Ваш домашний каталог, поэтому делать это нужно от имени root.

Если Вы выполняете копирование на жесткий диск, отформатированный под файловую систему Windows, воспользуйтесь командой **tar**, чтобы сохранить атрибуты файлов.

```
cd /home
sudo tar czf /media/disk/home.tar.gz .
```

Она скопирует все содержимое Вашего раздела **/home**, сжав его с помощью *gzip*, в файл **home.tar.gz** на Вашем внешнем жестком диске, примонтированном к **/media/disk**. Точка (.) в конце команды дает команде **tar** указание архивировать текущий каталог, **/home**. Сжатие часто ускоряет процесс копирования, потому что при этом через шину USB передается меньше данных. Если у Вас медленный компьютер, можете пропустить шаг со сжатием, скомандовав

```
tar cf /media/disk/home.tar .
```

Когда Ваши данные скопируются, можно будет удалить содержимое **/home**, но оставить сам каталог, а затем воспользоваться командой **df**, чтобы посмотреть, сколько пространства теперь занимает Ваш корневой раздел:

```
df -h /
```

Это подскажет вам требуемый размер корневого раздела. Добавьте 30–50 % к текущему размеру, чтобы обеспечить достаточный объем свободного дискового пространства «на вырост» и избежать фрагментации файловой системы. Теперь отмонтируйте Вашу корневую файловую систему, которую вы должны были примонтировать, чтобы скопировать оттуда файлы. Загрузитесь с Live CD и запустите менеджер разделов (Ubuntu использует *GParted*), чтобы установить новый размер корневой файловой системы, а затем создайте новую файловую систему для **/home** на свободном месте, которое Вы только что выделили. Перераспределение данных потребует времени, и в его процессе пользоваться компьютером нельзя.

Когда дело будет сделано, перезагрузите компьютер, чтобы он считал новую таблицу разделов. Примонтируйте куда-нибудь новый раздел и скопируйте файлы обратно с Вашего внешнего диска – либо через файловый менеджер, либо распаковав ранее созданный архив командой

```
tar xf /mnt/disk/home.tar.gz -C /mnt/newhome
```

Опция **-C** (заглавное) указывает **tar**, куда поместить файлы. За ней следует путь, по которому примонтирован новый раздел. Заключительный шаг – редактирование файла **/etc/fstab** на Вашем корневом разделе. Добавьте в него запись для **/home**:

```
/dev/sdaN /home ext3 defaults 0 0
```

Замените **sdaN** на имя созданного вами раздела, а **ext3** – на тип выбранной вами файловой системы (если она другая). Теперь перезагрузите компьютер без Live CD. Все должно будет работать, как и раньше, но уже при отдельном разделе **/home**.

Если у Вас нет внешнего жесткого диска, ситуация немного осложнится. Определите, сколько свободного пространства Вам потребуется для хранения данных на домашнем разделе, командой

```
du -h /home
```

и воспользуйтесь Live CD, чтобы сжать корневой раздел на величину, чуть большую объема сво- »

бодного пространства, который Вы только что определили, и создать временный раздел в конце диска.

Этот новый раздел послужит Вам вместо внешнего жесткого диска в вышеописанных инструкциях. Когда Вы создадите новый раздел **/home**, он должен располагаться на дисковом пространстве, смежным с временным разделом. Покончив с настройкой и убедившись в том, что все работает правильно, Вы сможете снова загрузиться с помощью Live CD, удалить временный раздел, а освободившееся пространство выделить разделу **/home**.

Однако будьте настороже: при работе с файловыми системами всегда существует риск случайно отформатировать не тот раздел, или что в момент передачи данных произойдет сбой питания. **НБ**

4 Случайные сбои

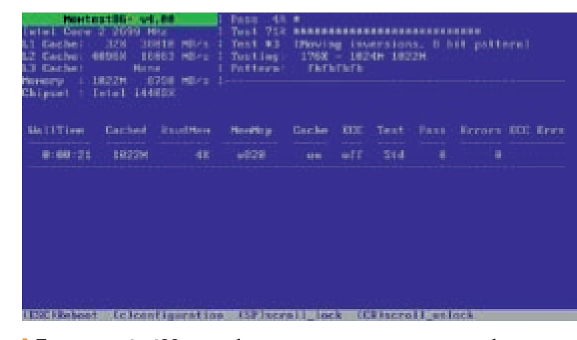
В моем Ubuntu происходят случайные сбои, не связанные ни с каким конкретным приложением. В версии 8.04 проблем не было, в 9.04 сбои случались иногда, а в 9.10 они зачастили, как в старые добрые времена Windows. Моя материнская плата — Intel D845GLVA с процессором Celeron 2,40 ГГц, i686. ОЗУ — 512 МБ, с возможностью расширения до 2 ГБ. Подозреваю, что в последних релизах Ubuntu есть ошибки управления памятью — а может, в последних версиях ядра. Не подскажете ли вы, в чем дело?

Эд Клери [Ed Clary]

Вы не сообщили, продолжают ли нормально работать релизы 8.04 и 9.04 (если предположить, что Вы их сохранили) и работают ли на этом компьютере другие дистрибутивы Linux или другие ОС. Но все равно, сказанное вами наводит на мысль об аппаратной проблеме. Когда крах системы возникает случайным образом и не выглядит связанным с каким-то конкретным ПО, то подозрение падает, в первую очередь, на оборудование, особенно если проблема со временем усугубляется, как у Вас. Имеют ли Ваши сбои между собой нечто общее – скажем, все они происходят при запуске приложений, интенсивно потребляющих ресурсы процессора?

Три основные причины аппаратных сбоев – перегрев, питание и память. Для начала вскройте корпус Вашего компьютера и проверьте, все ли вентиляторы работают, включая вентилятор блока питания. Затем посмотрите, не блокирует ли какая-либо деталь воздушный поток. Не нуждаются ли в чистке ребра радиатора? Для этого Вам очень пригодится баллончик со сжатым воздухом. Пока корпус открыт, заодно проверьте, правильно ли установлены в слотах карты PCI или VGA. Если их извлечь и снова установить, Вы не только убедитесь, что карты установлены правильно, но заодно и почистите контакты.

Проверить память несложно, но это требует времени. Большинство дистрибутивов на Live CD имеют опцию, позволяющую запускать программы тестирования памяти *memtest86* или *memtest86+*. Если у Вас еще нет под рукой ни одного такого дистрибутива, то хорошим выбором будет SystemRescueCD (сайт http://www.sysresccd.org). Загрузитесь с Live CD и введите **memtest** в командной строке загрузчика. *Memetest86* – самостоятельная программа, и ее нельзя запустить из-под работающей ОС. Пусть она проработает как можно дольше, как минимум два полных цикла. Еще лучше запустить тест на всю ночь, потому что ошибки памяти бывают нестационарными (это одна из причин случайной природы сбоев).



› Пусть *memtest86* проработает как можно дольше, чтобы выявить возможные причины загадочных сбоев.

5 Сертификация LPI

В Последние четыре года я работаю с Linux, в настоящий момент — Slackware. Я хотел бы сдать экзамены LPI (Linux Professional Institute), но не знаю, где искать. Экзамен должен сдаваться онлайн или в каком-нибудь колледже? Я просмотрел кое-какую документацию в Интернете, но не уверен, который путь лучше.

Не подскажете ли вы мне, какой вариант будет наилучшим, и где сдавать экзамен? Буду благодарен за любой совет.

Пит Клоу [Pete Clough]

Вы хотите прослушать профессиональный учебный курс, получить информацию о том, где в сети можно найти справочные материалы или просто узнать, где проходить тестирование? Профессиональные учебные курсы предоставляют многие компании. Найдите ближайшую к себе, пошарив в Web, или зайдите на сайт LPI (http://www.lpi.org) и просмотрите список официальных партнеров по обучению. Кроме того, ознакомьтесь с учебными планами и ценами; прикиньте, сколько с Вас вычтут, если Вы не сдадите экзамен с первого раза и потребуется дополнительное обучение для повторной попытки; и тогда принимайте окончательное решение.

Если Вы намерены заняться самообразованием, рекомендую книгу издательства O'Reilly «LPI Linux Certification in a Nutshell [Краткое описание сертификации LPI]», с подзаголовком «A Desktop Quick Reference [Быстрый справочник по настольной системе]». Оба названия не совсем верны, потому что, прежде чем браться за чтение, придется потренироваться еще и в тяжелой атлетике – в книге около 1000 страниц. Мы высоко ценим ее – она рассказывает о каждом из модулей и описывает все, что нужно знать. Кроме того, в ней приведены примеры тестовых вопросов, иллюстрирующие характер экзамена.

Экзамены принимаются в ряде центров тестирования. На сайте LPI есть вся информация по поиску центра сертификации и записи на экзамены. LPI приводит и примерные цены за экзамен – обычно это местный эквивалент суммы в $160. Поэтому остерегайтесь организаций, которые запрашивают за один только экзамен суммы, существенно превышающие названную. Но если Вам предлагается не только сдача экзамена, но и обучение, особенно интенсивный курс, платить придется много больше.

Я бы настоятельно рекомендовал первым делом обзавестись книгой O'Reilly, чтобы понять, какой уровень требуется, и оценить собственный. Затем можно будет принять обоснованное решение о том, как «подтянуть» свои знания. Наконец, загляните в **LXF110** на wiki.linuxformat.ru – там есть статья «Стать дипломированным Linux-гуру» с информацией и об экзаменах LPI, и о многом другом.

Другим «виновником» случайных сбоев может быть блок питания, особенно если у Вас дешевая модель от безымянного изготовителя. Нестабильность напряжения, подаваемого на Вашу систему, может не только приводить к случайным сбоям, но и сократить срок службы остального оборудования. Простого способа проверить работу блока питания, не прибегая к его замене, нет. Но можно направить на вентиляционные отверстия струю сжатого воздуха из баллончика, чтобы проверить, не нарос ли на нем слой пыли. Если у Вас есть еще один компьютер, можно попробовать взять блок питания от него. В противном случае, для проверки останется только замена на новый блок. Поэтому я и указал данный вариант последним, хотя и не считаю, что блок питания – наименее вероятная причина сбоев. **ГМ**

6 Порты приписки

В LXF128 я, как всегда, с удовольствием прочел рубрику «По рецептам доктора Брауна», и там было сказано то, о чем я не подозревал — а именно, что исходящий IP имеет случайный номер порта. Почему это так? Я имею в виду — независимо от NAT?

Дж. Г. Оуэн [JG Owen]

Чтобы установить соединение, программа должна использовать свободный порт. Если Вы посылаете запрос web-страницы на порт 80, и при этом у Вас работает еще и web-сервер, то две программы будет прослушивать один и тот же порт. Для этого даже не надо запускать web-сервер – достаточно просто открыть два браузера.

Поэтому программа выбирает порт, который на данный момент не используется, и отсылает свой запрос на указанный порт сервера (для HTTP это будет порт 80). Так как запрос включает и IP-адрес, и порт для ответов, сервер точно знает, куда следует отправлять данные.

Причина использования портов с высокими номерами в том, что порты с номерами ниже 1024 присваиваются конкретным протоколам и могут быть открыты только с использованием привилегий root. Эта мера предпринята для усиления безопасности, чтобы затруднить захват портов и перехват трафика для шпионского ПО. **ПХ**

7 Samba упрямится

В Мне никак не решить проблему на новом офисном компьютере с Mandriva Linux 2010.0. Я — один из шести пользователей LAN. На данный момент я должен загружаться и вручную монтировать сетевые ресурсы на сервере, но вообще-то использую закладку с *Smb4k* — так быстрее. Я хочу, чтобы сетевой ресурс монтировался автоматически во время загрузки. Я поместил в /root/.smbcredential команды

```
username=мое_имя_пользователя
password=мой_пароль
```

и установил права доступа в 0700. Кроме того, я приписал к /etc/fstab **следующую строку:**

```
//192/168/0/1/linux /home/uid/sharename
smbfs credential=/root/.
smbcredentials,dmaskw7,fmaskw7 0 0
```

Я предполагал, что URL — это статический адрес сервера, что мой разделяемый ресурс будет заменен на /linux, **и что** /home/uid/sharename **— точка монтирования.**

Алан Секер [Alan Secker]

О Подробно описав все, что сделали Вы сами, Вы не сказали, что происходит при попытке примонтировать сетевой ресурс. От имени root скомандуйте

```
mount /home/uid/sharename
```

– либо подмонтируется сетевой ресурс, либо выведется сообщение об ошибке, которое и послужит ключом к поиску причины проблемы. Если сообщение будет выглядеть как

```
'mount: wrong fs type, bad option, bad superblock
on //192.168.0.1/linux'
```

то причина в том, что Ваш дистрибутив больше не поддерживает не рекомендуемую к употреблению файловую систему *smbfs*, вместо которой следует использовать *cifs*.

В противном случае, неудача с монтированием может вызваться использованием IP-адреса сервера вместо его имени **netbios**. Совместный доступ к файлам в сети Windows основан на именах, которые не обязательно совпадают с именем Вашего хоста Linux. Имя netbios показано в первом столбце в сетевом окружении (Network Neighborhood) в *Smb4k*.

Замените строку в файле **/etc/fstab** на такую:

```
//ИМЯ_СЕРВЕРА/linux /home/uid/sharename
cifs credentials=/root/.smbcredentials…
```

и измените права доступа для файла с регистрационной информацией на 600 (это не должен быть исполняемый файл). Теперь попробуйте примонтировать ресурс через терминал. Если и это не сработает, и Вы не получите более никаких полезных сообщений об ошибках, просмотрите файлы журналов на сервере, чтобы проверить, получает ли он запрос на монтирование ресурсов и позволяет ли он сделать это. **МС**

8 Отказ двойной загрузки

В Недавно я приобрел LXFS19 *Get Started With Ubuntu*. Я выделил весь внешний диск USB для Ubuntu, с тем, чтобы не трогать имевшийся на ПК раздел Windows Vista (по крайней мере, так это выглядело). Моя проблема в том, что теперь я не могу загрузить Vista (ошибка *Grub* 21), не подключив мой внешний диск USB, после чего я получаю выбор загрузки либо Ubuntu, либо Vista.

Это препротивная ситуация, потому что приходится дожидаться появления меню двойной загрузки, чтобы загрузить Vista. А иначе по умолчанию загружается Ubuntu. Кроме того, меня беспокоит, что если что-то случится с внешним диском, я не смогу загружать не только Ubuntu, но и Vista.

Барт Вердучи [Bart Verduci]

О Ошибка *Grub* 21 означает «Selected disk does not exist [Выбранный диск не существует]». Установщик Ubuntu записал каталог *Grub* на Ваш внешний диск, а загрузочный код – в MBR (Master Boot Record) на внутреннем жестком диске. А Вам надо, чтобы на внутреннем жестком диске был весь загрузочный код – тогда система не станет искать его на внешнем диске, кроме тех случаев, когда Вы выберете Ubuntu из загрузочного меню. Сделать это можно, создав конфигурацию, в которой загрузчик Windows будет вызывать *Grub*. Альтернатива – переместить все файлы, нужные *Grub*, на внутренний диск. Последний вариант проще в осуществлении.

Grub умеет работать с большинством файловых систем – за одним, но выдающимся исключением: NTFS. Это значит, что код *Grub* нельзя переместить на Ваш основной раздел Windows. Вместо этого Вам нужно немного сжать Ваш основной раздел Windows (C:), чтобы создать еще один небольшой раздел и отформатировать его под FAT (можно использовать и файловую систему Linux, но благодаря FAT раздел будет доступен и из-под Windows).

Сделать это можно и с помощью программы наподобие *PartitionMagic* из-под Windows, или воспользоваться *Partition Editor* под Ubuntu (это безопаснее). Если опции *Partition Editor* нет в меню Administration [Администрирование], установите *GParted* через *Synaptic*. Раздел, который Вам нужно создать, должен быть небольшим – 3 МБ на него более чем достаточно. Создайте раздел, отформатируйте его, и он появится в меню Places [Переходы] (если этого не произойдет, потребуется перезагрузиться).

Примонтируйте этот раздел, а затем через терминал скопируйте туда нужные файлы, командами

```
sudo mkdir /media/disk/boot
sudo cp -a /boot/grub /media/disk/boot
```

Они создадут загрузочный каталог на новом разделе (здесь предполагается, что он примонтирован как **/media/disk**) и копируют туда содержимое »

Часто задаваемые вопросы

Мобильная широкополосная связь

» Что такое мобильная широкополосная связь, про которую все болтают? Это разновидность ADSL поверх соединения через мобильный телефон?

Вроде этого. Она использует мобильную телефонную сеть 3G, но технология ее работы – не ADSL, а HSPA (High Speed Packet Access), разработанная специально для мобильного применения. Она используется смартфонами, которым требуется постоянное (или, как минимум, частое) подключение к Интернету.

» И это даст мне скорость широкополосной связи?

Нет, это не есть широкополосная связь в исходном смысле этого слова (как и некоторые более медленные соединения по наземным проводным линиям), но ее скорость достаточна для задач типа чтения электронной почты или просмотра видео на YouTube. Провайдеры наперебой предлагают всякие скорости, но реально они зависят от мощности сигнала и сопутствующих факторов.

» Что мне нужно, чтобы получить мобильную широкополосную связь?

Широкополосный модем и учетная запись в одной из компаний, занимающихся широкополосными мобильными коммуникациями. Большинство из них, кстати, модем предоставляют, но можно обойтись и своим. Почти все телекоммуникационные компании используют один и тот же модем, в виде USB-брелка, требующий SIM-карты для аутентификации соединения через вашу учетную запись.

» А сколько все это будет стоить?

Как и в большинстве подобных случаев, цена зависит от провайдера, продолжительности соединения и объема передаваемых данных. Кое-кто предлагает даже безлимитные тарифы или пакеты трафика.

» Хорошо ли мобильная широкополосная связь ладит с Linux?

В настоящее время – очень хорошо. Похоже, все компании предоставляют модемы от одного и того же производителя, Huawei. Этот модем имеет драйверы для ядра Linux, так что он должен «просто работать». Вам потребуется настроить модем для соединений по коммутируемым каналам, используя *KPPP* или *Gnome-PPP*. Модем будет виден как **/dev/ttyUSB0**. Поскольку он использует для аутентификации SIM-карту, в качестве входного имени и пароля можно указывать что угодно (хотя ПО и предполагает, что вы должны ввести эти данные).

» Имеет ли значение используемый дистрибутив?

Стандартные настройки PPP должны работать с любым дистрибутивом. Дистрибутивы, применяющие *Network Manager*, имеют преимущество, поскольку он теперь обеспечивает поддержку 3G-модемов. В таком дистрибутиве при первом подключении модема может появиться окно, предлагающее выбрать провайдера, после чего вся настройка выполнится автоматически. А затем для включения и отключения модема вы сможете пользоваться меню *Network Manager*.

каталога **/boot/grub** из Ubuntu. При условии, что из Ubuntu Ваш диск Windows виден как **sda**, а внешний диск – как **sdb**, сообщить *Grub* о новом каталоге, где искать загрузочные файлы, можно так:

```
sudo grub-install /dev/sda
```

Если порядок следования дисков другой, отключите внешний диск, перезагрузитесь с CD, содержащего дистрибутив Ubuntu, и дайте ту же команду. Это гарантирует, что *Grub* будет ссылаться только на файлы, находящиеся на внутреннем диске.

Теперь нужно предпринять еще один шаг, поскольку *Grub* по-прежнему по умолчанию загружает Ubuntu и будет пытаться это делать даже при отключенном внешнем диске. Откройте файл **boot/grub/menu.lst** на диске Windows и посмотрите на строку, соответствующую системе по умолчанию. Скорее всего, там будет

```
default 0
```

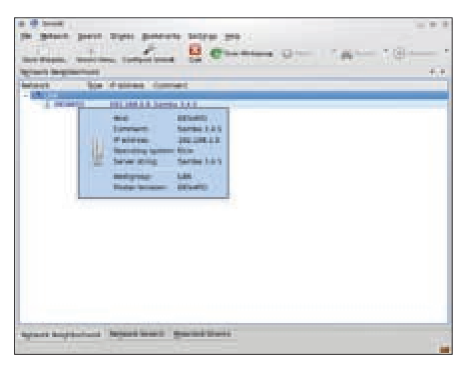
что устанавливает первую опцию меню как используемую по умолчанию (нумерация в *Grub* начинается с 0). Установите в этой строке **1** вместо **0**, чтобы по умолчанию выбиралась вторая опция меню (или укажите номер, соответствующий позиции Windows в меню загрузки). Сохраните файл. Теперь *Grub* по умолчанию будет загружать Windows.

Если менеджер обновлений Ubuntu когда-нибудь установит новое ядро, будет изменено и содержимое каталога **/boot/grub**. Тогда Вам потребуется еще раз скопировать его на диск Windows. **НБ**

9 Неформат

В **я был бы очень признателен, если бы вы осветили на страницах журнала правильную процедуру сборки плейера *Amarok* в Ubuntu с поддержкой всех кодеков. Его, конечно, можно скачать из официальных источников в виде готового двоичного пакета, но за это придется «заплатить» поддержкой AAC. Как получить ее?**
Влад

О Будем исходить из предположения, что речь идет о достаточно свежей версии дистрибутива Ubuntu (инструкции для более ранних версий доступны по адресу https://help.ubuntu.com/community/RestrictedFormats). Для поддержки большинства проприетарных форматов сайт Ubuntu рекомендует установить метапакет *ubuntu-restricted-extras* из секции multiverse основного репозитория. Кроме того, в среде рабочего стола Gnome дистрибутива Ubuntu реализована установка кодеков по запросу. Достаточно попытаться открыть файл AAC в *Nautilus*, и система сама предложит инсталлировать пакет *gstreamer0.10-plugins-bad* (для Karmic Koala) из секции universe. После этого плейеры, использующие *GStreamer*, получат поддержку AAC. Вы же используете *Amarok* (к слову, достаточно странный выбор для дистрибутива на базе Gnome) – он не имеет отношения к *GStreamer*, и необходимый кодек придется установить вручную (в KDE это произошло бы автоматически). Достаточно просто добавить пакет *libxine1-ffmpeg* из секции main и перезапустить плейер. Стоит отметить, что для поддержки большинства проприетарных форматов в KDE достаточно установить метапакет *kubuntu-restricted-extras* из секции multiverse. Обратите внимание, что распространение данных программ в вашей стране может быть ограничено патентами или законами. Именно поэтому их не включают в установку по умолчанию. **ЕК**



› **Использование *Smb4k* для поиска и подключения сетевых ресурсов Windows дает информацию, позволяющую монтировать их автоматически.**

10 А как же свобода слова?

В **Обновил свою систему (Mandriva 2009.1) до версии 2010.0 (рабочий стол KDE4) и вдруг обнаружил, что не могу установить *Skype* — в списке доступных программ пакета с таким именем нет. Подскажите, что делать?**
Виталя Цегенько

О *Skype* относится к проприетарным программным продуктам, и о его включении в репозитории Mandriva до сих пор идут жаркие споры. Выигрывает, увы, сторона «используйте свободную альтернативу», поэтому найти *Skype* в официальной версии дистрибутива можно только в коммерческом репозитории Mandriva PowerPack. Но не все потеряно – программа есть в неофициальном репозитории итальянского сообщества MiB. Вам достаточно подключить его с помощью сервиса urpmi.mandriva.ru и установить Skype командой:

```
urpmi skype
```

Вы также можете просто скачать и установить пакеты по следующим ссылкам:

» для 32-битных систем на основе Mandriva 2010 – http://mib.pianetalinux.org/MIB/2010.0/32/basic/skype-2.1.0.81-69.1mib2010.0.i686.rpm -

» для 64-битных систем: http://mib.pianetalinux.org/MIB/2010.0/64/basic/skype-2.1.0.81-69.1mib2010.0.x86_64.rpm. **АК**

Коротко про...

Поиск файлов

В Linux есть два основных инструмента для поиска файлов: *locate* и *find*. Первый работает по принципу хранения базы данных файлов, имеющихся в Вашей системе. Это быстрый инструмент, но его возможности ограничиваются только файлами, присутствовавшими на диске на момент последнего обновления базы данных. Большинство дистрибутивов устанавливают скрипт *Cron* для поддержания базы данных в актуальном состоянии. Команда *Locate* также ограничена поиском по именам файлов; ее вид –

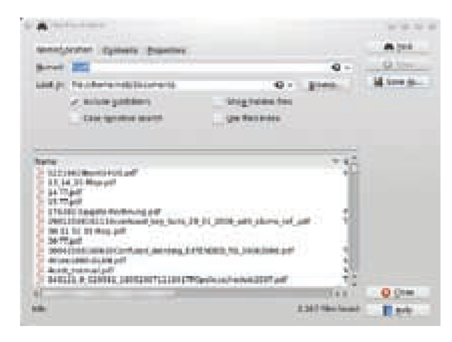
```
locate имяфайла
```

Чтобы поиск не зависел от регистра, добавьте ключ **-i**.

Альтернатива – команда *find*, выполняющая поиск непосредственно по файловой системе. Она работает медленнее и может искать только в тех областях, к которым пользователь имеет доступ с правом чтения, зато предоставляет самую современную и актуальную информацию. Она позволяет указывать каталоги, где требуется искать, и каталоги, которые следует исключить из поиска, а также перечислять свойства, отличные от имени файла – например, дату создания файла или имя его владельца:

```
find -name '*имяфайла*'
find /usr -iname '*имядругогофайла*'
find /usr -maxdepth 2 -iname '*какойфайл*'
```

Первая из этих команд начинает поиск в текущем каталоге и затем спускается вниз по иерархии, просматривая все подкаталоги. Вторая команда начинает с каталога **/usr** и выполняет поиск, нечувствительный к регистру. Третья команда делает то же самое, но спускается вглубь только на два каталога.



› **Инструмент поиска *KFind* — это команда *find* «с человеческим лицом».**

Благодаря дополнительным опциям, *find* – гораздо более гибкий инструмент. Зато команда *locate* великолепна для быстрого поиска. Ее вывод можно перенаправить на вход команды *grep*, чтобы выполнить поиск по конкретным каталогам:

```
locate -l myfile | grep /home/
```

Учтите также, что *locate* ищет подстроки, а *find* – точные совпадения, отсюда и использование * как символа шаблона.

Большой вопрос Как синхронизировать мои домашние каталоги?

11 Чувство синхронности

В У меня есть настольный компьютер и ноутбук. Оба они работают под Linux, но в разных дистрибутивах. Хотелось бы синхронизировать их домашние каталоги, чтобы последние версии важных файлов имелись в обеих системах. Догадываюсь, что помочь тут может *rsync*, но этот вариант выглядит очень сложным, и man-страница ничуть его не облегчает. Кроме того, жесткий диск на настольном компьютере намного больше, чем на ноутбуке, и на ноутбуке все файлы просто не поместятся, так что мне нужна возможность исключить некоторые файлы из процесса синхронизации. Что вы порекомендуете? Все же применять *rsync*? А раз так, нет ли для нее более дружелюбного интерфейса?

Роджер Кэмерон [Roger Cameron]

О Вы правы, думая об *rsync*, но в Вашем случае у него есть недостаток: это односторонний протокол. Для компьютеров А и В *rsync* в состоянии копировать или с А на В, или с В на А. И если Вы хотите модифицировать файлы на обоих компьютерах, *rsync* Вас не спасет. Лучшая из программ, решающих Вашу задачу, называется *Unison* (http://www.cis.upenn.edu/~bcpierce/unison). Это инструмент двусторонней синхронизации: если Вы редактировали файл **foo.txt** на компьютере А, а файл **bar.png** – на компьютере В, оба файла обновятся на обоих компьютере. Если Вы модифицируете один и тот же файл в двух местах, программа спросит, какая из версий является актуальной.

При первом запуске *Unison* спросит, какие каталоги синхронизировать. Первый из них должен быть локальным, а второй – находиться на удаленном компьютере, с доступом через SSH. Поскольку Вам требуется синхронизировать домашние каталоги, Вы можете оставить эти поля пустыми, указав во втором случае только имя хоста. Кроме того, можно указать имена пользователей, если на синхронизируемых компьютерах они разные. При первом запуске *Unison* потребуется некоторое время на сканирование обоих каталогов, а затем Вы сможете указать, какие директории и файлы Вы хотите игнорировать. В процессе первой синхронизации большинство различающихся файлов будут помечены как конфликтующие, потому что Unison еще не известно, который Вам нужен – укажите его клавишами-стрелками влево/вправо. При последующих синхронизациях дело пойдет быстрее.

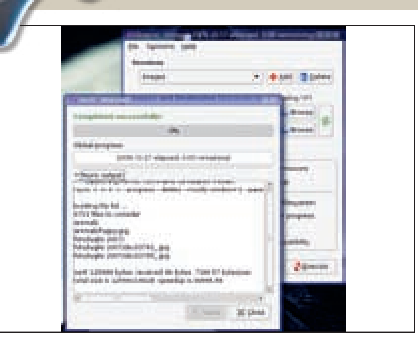
Настройки для каталогов, хостов и путей, исключаемых из синхронизации, сохранятся в профиль, по умолчанию в **~/.unison**. При желании создать и другие профили синхронизации, ознакомьтесь с форматом этого файла. Чтобы заставить *Unison* использовать нужный профиль, укажите его имя как аргумент или выберите имя профиля в диалоговом окне. *Unison* может работать и из командной строки – ее можно использовать в скриптах автоматической синхронизации.

Unison применяет для связи между компьютерами SSH, поэтому при каждой очередной синхронизации программа будет запрашивать пароль пользователя для входа на другой компьютер. Чтобы избежать этого, установите ключи SSH. Запустите на обоих компьютерах команду **ssh-keygen** и примите ответы по умолчанию для всех задаваемых вопросов, но не присваивайте ключу парольной фразы [passphrase]. Создадутся два файла в каталоге **~/.ssh**, с именами **id_dsa** и **id_dsa.pub**. На обоих компьютерах введите

```
ssh другой_компьютер cat .ssh/id_dsa.pub >>~/.ssh/authorized_keys
```

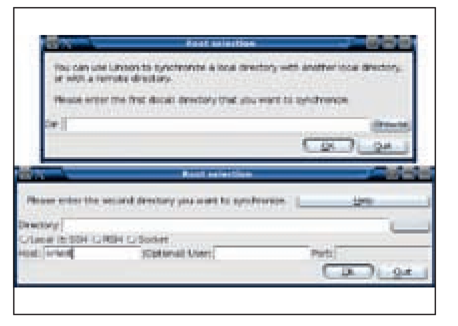
Эти команды добавят публичный ключ SSH другого компьютера к списку пользователей, которые могут подключаться автоматически. **НБ**

Шаг за шагом: Синхронизируемся в Unison



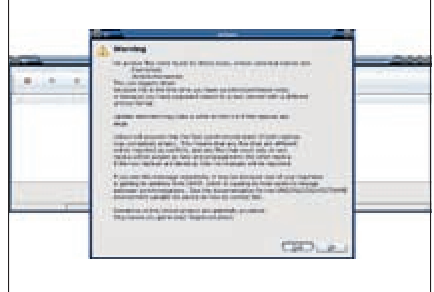
1 В одни ворота

Хотя для *rsync* и имеются клиенты с графическим интерфейсом, например, *Grsync* (http://www.opbyte.it/grsync), протокол у них все равно односторонний.



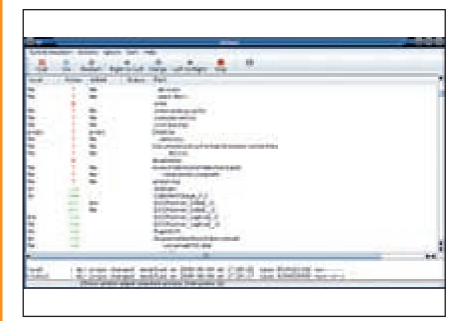
2 Выбор пути

Unison запросит пути исходного и целевого каталогов; для домашних директорий их можно оставить пустыми. Укажите только имя удаленного хоста.



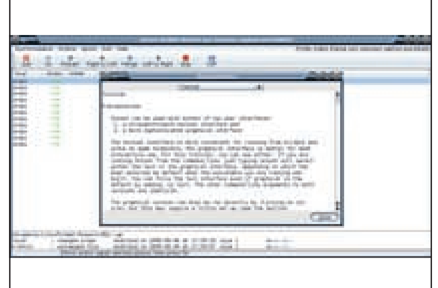
3 Понадобится время

Unison выполнит начальное сканирование домашних каталогов. При первом запуске оно довольно долгое, но впоследствии будет проходить быстрее.



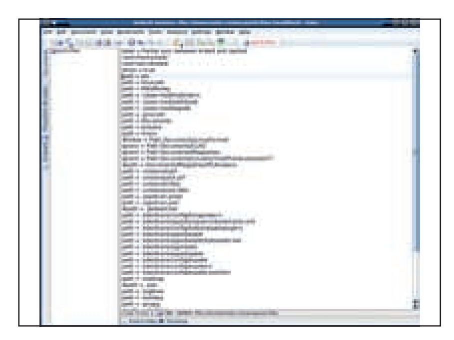
4 Дайте совет

При первом сканировании *Unison* выдаст ряд вопросов, поскольку еще не знает, какую из версий файла вы хотите сохранить, и предоставит решение вам.



5 Встроенная помощь

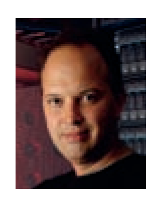
Вместо обычных man- и HTML-страниц *Unison* имеет встроенную систему помощи, структурированную по темам, включая удобную пошаговую инструкцию.



6 Подробные профили

Вы можете создать профиль, указывающий, какие каталоги синхронизировать. Использование профилей ускорит процесс. LXF

LXFHotPicks



Ник Вейч
Ник компилирует Hotpicks, используя текстовый редактор и головной мозг. А иногда — только текстовый редактор.

Программа для аннотации видео

Advene

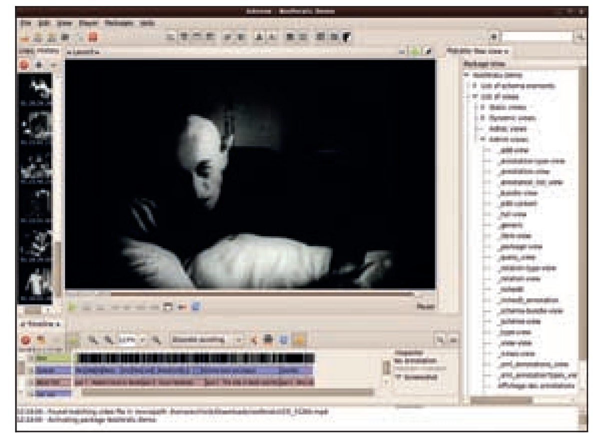
Версия 0.40 **Сайт** http://liris.cnrs.fr/advene/index.html

Google, наверное, прав, убеждая весь мир в том, что совместно работать над документами можно – и даже более эффективно – через сеть, но что если проект, над которым вы трудитесь, не ограничивается простым текстом? Стандартный обменный курс слов на картинки показывает, что художественный фильм соответствует 135 миллионам слов (90 минут×60 секунд×25 fps×1000 слов), а это малость побольше, чем в вашей обычной порции HotPicks, если вы вообще можете себе такое представить.

Advene был разработан, чтобы удовлетворить потребность студентов, изучающих кинематографию, и, по случаю, также тех, кому захочется или понадобится создавать комментарии или заметки к видеофайлам. Такое бывает, если надо изучить и задокументировать тему, отметить испорченные фрагменты или добавить собственные забавные надписи к классическим сценам.

Advene создает пакет всех этих аннотаций, заметок и маркеров, а также ссылку на исходный видеофайл – в конце концов, ваша копия «Носферату» не обязательно совпадает покадрово с моей. Затем кто угодно может загрузить этот пакет и открыть совместно с данным видеофайлом, изменить и дополнить ваши записи и внести свои (*Advene* отслеживает авторство).

Ваши возможности не ограничиваются исключительно текстовой анимацией – имеется, например, поддержка наложения SVG, и вы можете легко обвести кружком или заменить часть изображения собственным корявым рисунком.



› Майк сказал, что идет за кружкой, и мы решили, что это про вкусное пиво…

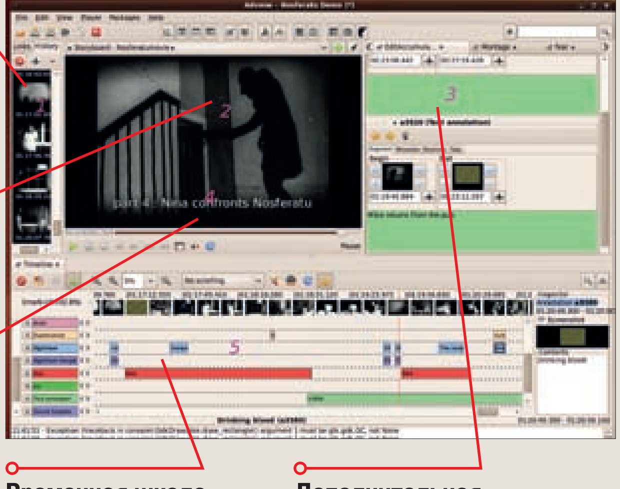
Я люблю кино

Открывает список недочетов пользовательский интерфейс. Возможно, это отчасти придирка – ведь он сумел уместить на экране немало информации; но изначально интерфейс весьма сложен для понимания того, что и где вообще происходит и как это можно изменить. Шкала времени внизу экрана отображает слои текущей аннотации или треки, и помимо этого вы видите информацию на панелях справа и слева от окна просмотра. Простые аннотации можно накладывать на экран просмотра, но более подробные заметки лучше отображаются на одной из этих панелей. По умолчанию слева отображается ваша история посещений в виде миниатюр, что удобно для перекрестных ссылок.

Если вы хотите понять, как работает *Advene*, скачайте с сайта обучающие ролики. Это просто фантастические фильмы, и вы получите от процесса учебы большое удовольствие.

А я закругляюсь: надо писать аннотации к шоу, которое я называю «Британия нашла Пирса Моргана, но все предложения будут рассмотрены» [в оригинале обыгрывается название популярного шоу «Britain's Got Talents» – «Британия ищет таланты» и имя его ведущего, Пирса Моргана [Piers Morgan], – *прим. пер.*].

Исследуем интерфейс Advene



История посещений в виде миниатюр
Этот удобный орган управления отслеживает, где вы побывали в видеофайле.

Главное окно
Здесь воспроизводится ваше видео.

Ваши мудрые изречения
Если перекрывающие надписи достаточно короткие, они появится на экране.

Временная шкала
Цветные полоски визуально отмечают наиболее интересные фрагменты видео.

Дополнительная информация
Всплывающие окна и прочие данные могут появляться на боковых панелях.

Индикатор прогресса

Gprog 1.10

Версия 1.10 **Сайт** http://stromberg.dnsalias.org/~strombrg/gprog

Одно из главных удовольствий в жизни – найти небольшой участок общей схемы, который можно деликатно подправить, просто чтобы вплести в ковер повседневности цветные нити чуда. Ну, или еще чего-нибудь. Иными словами, иногда от малых причин проистекают большие последствия, и именно шансами вбросить в него массы милых мелочей и хорош открытый код.

Чересчур-то не обнадеживайтесь – *Gprog* не ликвидирует угрозу глобального потепления и даже не расскажет, что у вас будет на ужин; зато он может сообщить вам, сколько времени продлится этот ужин. *Gprog* – это графический индикатор прогресса для канальных операций (типа копирования файлов или чего-то подобного), с уймой опций, чтобы показать вам, сколько данных должно быть передано, какова средняя скорость передачи данных, сколько времени уже длится процесс и когда ожидать его окончания.

По своей природе подобные процессы мониторинга подвержены принципу неопределенности Гейзенберга – т.е., сам факт наблюдения за передачей данных искажает картину. Используя многопроцессорный модуль Python, *Gprog* может задействовать дополнительные ядра, чтобы отделить операцию копирования от наблюдения за копированием и, таким образом, сохранить и полную производительность, и достаточно точный отчет о состоянии процесса. Если автор *Gprog* когда-нибудь придет к вам в гости на ужин, ризотто зря не пропадет.

Как уже упоминалось, *Gprog* – приложение Python. Оно самодостаточно, и в данном случае не требует сверх имеющегося в стандартно поставляемых пакетах Python ничего, кроме *PyGTK* (в зависимости от вашего дистрибутива). Вызовите его из командной строки, чтобы увидеть параметры использования, или запустите тестовый режим (который копирует файлы из **/dev/zero**), чтобы ознакомиться с графическим интерфейсом через предпросмотр.

› **Уже все? Уже все? Уже все...**

> «Gprog — индикатор прогресса операций типа копирования.»

Программа для чтения электронных книг

FBReader 0.12.7

Версия 0.12.7 **Сайт** www.fbreader.org

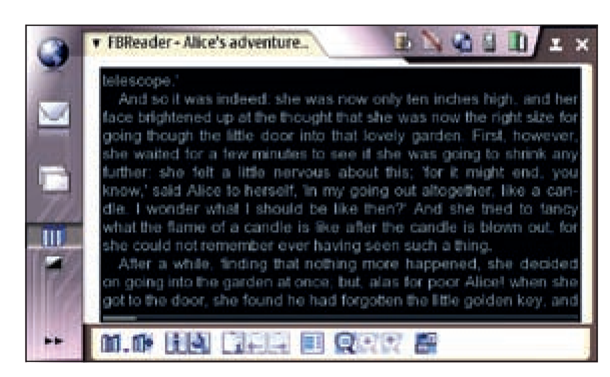
Устройство Kindle от Amazon открыло миру глаза и дало толчок идее чтения электронных книг в дороге; но многие предпочли бы читалку, которая не станет тайком удалять книги во мраке ночи, когда все мирно спят. И, возможно, поддерживала бы побольше форматов. Да чтобы была свободной…

Ну, устройство-то вам придется искать самим [правда, в СНГ поиск будет недолгим, – *прим. ред.*], зато программа *FBReader* более чем грамотно окажет вам содействие в чтении электронных книг в разнообразных форматах. Она работает на Linux и на многих устройствах, использующих Linux – например, Nokia N800/N810 с Maemo.

Поддерживаются форматы **.fb2**, **.epub** и Plucker e-book, применяемые большинством коммерческих и свободных ресурсов электронной литературы. Пока что *FBReader* не поддерживает просмотр PDF-файлов, и это изъян, поскольку данный формат также популярен среди электронных книг. Есть планы реализовать поддержку PDF в будущем, хотя вначале, скорее всего, для чисто текстовых файлов, а не документов, содержащих еще и изображения (что мы можем объяснить упором на экономию ресурсов и простоту использования). Тем не менее, обычные форматы электронных книг будут отображаться с картинками, если уж они там есть.

Маленькая и шустрая

На настольном ПК вы, скорее всего, предпочтете по-прежнему держаться *Evince* – все-таки это универсальная программа для чтения документов; но для мобильных устройств с Linux *FBReader* – идеальное приложение, благодаря его быстроте и низкому потреблению ресурсов. Для телефонов Android имеется порт *FBReader* на Java (креативно названный *FBReaderJ*), обладающий примерно тем же набором функций и доступный на том же сайте или в хранилище приложений Android.



› **Изначально разработанный для мобильных устройств, *FBReader* неплох и для чтения электронных книг на настольном компьютере.**

> «Поддерживаются форматы .fb2, .epub и Plucker.»

Тестирование служб DNS

Namebench

LINUX FORMAT Hottest Pick

Версия 1.2 **Сайт** http://code.google.com/p/namebench

Создавая Hotpicks, мы применяем для приложений список критериев, где нужно галочкой отметить ожидаемые качества, как то: программа должна быть интересной и полезной, она должна работать и, опционально, чем-то бесить Майка. Программам редко удается заполучить все четыре галочки, и случай с *Namebench* не стал исключением, ибо Майку программа тоже понравилась.

Ослепительно простая идея, на которой строится эта программа, заключается в проверке, можно ли улучшить производительность вашей работы в Сети. Интернет работает по странным принципам, и одна из самых дурацких концепций – это доменное имя. На деле никаких имен у компьютеров в Сети нет – есть одни номера; это те самые IP-адреса (ну ладно, все не так просто, но и не настолько сложно, как «Работа в сети для продвинутых нейрохирургов космических атомолетов»). Беда в том, что имена (а иногда и номера) в Сети могут блуждать, и чтобы связывать их вместе, предусмотрена служба Domain Lookup Service, или сервер имен (nameserver). Когда вы набираете www.linuxformat.ru в *Firefox*, происходит следующее: ваш компьютер соединяется со своим зарегистрированным сервером имен и запрашивает IP, связанный с этим именем. Затем сервер имен отправляет вам назад пакет информации с IP-адресом, и ваш браузер теперь может связаться с сайтом напрямую через Интернет.

Подвижные части

Сервис этот очень гибкий, но при работе в Сети способен создать эффект бутылочного горла. Скорость вашего сервера имен по умолчанию (обычно того, что поддерживается вашим провайдером) может заметно плавать. Помните, что, хотя обычно время отклика занимает миллисекунды, конкретная страница, которую вы просматриваете, должна реагировать на десятки обращений – в зависимости от того, хорошо ли запрограммирован сайт и сколько рекламы он содержит. Если сайт изобилует ссылками на графику, то время на разрешение имен хостов будет сравнимо со временем рендеринга изображений, а это безумие.

> «Скорость вашего DNS-сервера по умолчанию может варьироваться.»

› DNS интернет-провайдера может тормозить вашу работу. Испытайте его и ускорьте свою работу в Сети!

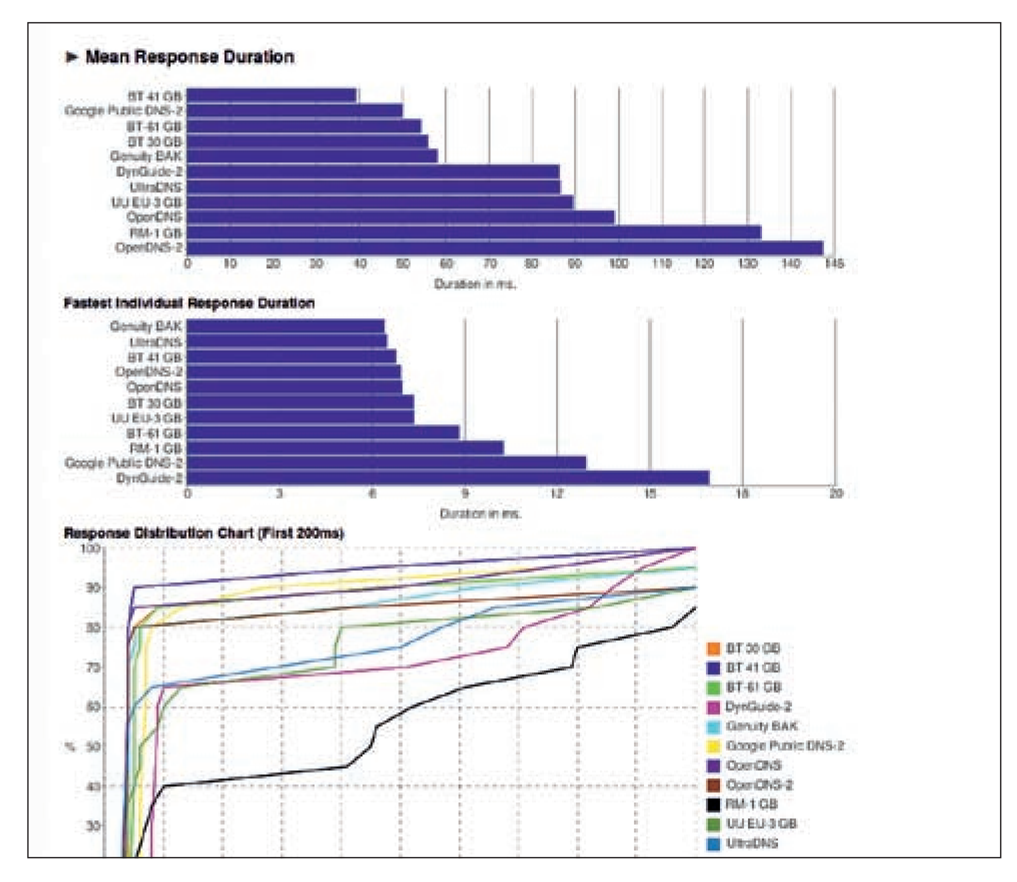
За последние годы в борьбу включилось множество серверов имен от третьих лиц, заявляясь на бóльшую скорость. Их выгода в данной области лежит в засылке вам рекламных объявлений (которые вы видели в тот раз, когда посетили несуществующий домен) или просто выяснении ваших сетевых пристрастий через ваш браузер, для последующего их употребления с добрыми/злыми намерениями. Сервисы вроде Google Public DNS и DynamicDNS объявляют о своем умении ускорить процесс просмотра, но верить ли рекламе?

Так или иначе, скорость каждого конкретного сервиса очень сильно зависит от того, в какой точке мира вы приобщаетесь к Сети и какие сайты пытаетесь открыть.

Что ведет нас к…

Цель *Namebench* – найти для вас наилучшие варианты, исследуя подборку известных доменов и дополнительно включая результаты поиска для сайтов, хранящихся в журнале вашего браузера. При запуске приложения возникнет окно настройки, куда можно будет ввести адреса DNS-серверов (уже настроенные включаются автоматически), количество тестов, которые вы хотите провести, и какие DNS-сервера надо включать. *Namebench* содержит список региональных DNS-серверов для разных частей земного шара (это одна из тех сфер работы в сети, где локальные ресурсы обычно намного лучше), и может добавить их к тесту.

Процесс тестирования может занять несколько минут, но в итоге перед вами (в вашем браузере) предстанет фантастический графический прорыв по производительности, а также оперативный отчет, подробно расписывающий, какой сервис подойдет вам больше всего. Честно говоря, результаты будут меняться каждый день, так что есть смысл установить *Namebench* и запускать проверку примерно раз в неделю, ради уверенности, что вы используете наилучшие для вас настройки.



HotGames Развлекательные приложения

Гонки на световых мотоциклах

Armegatron

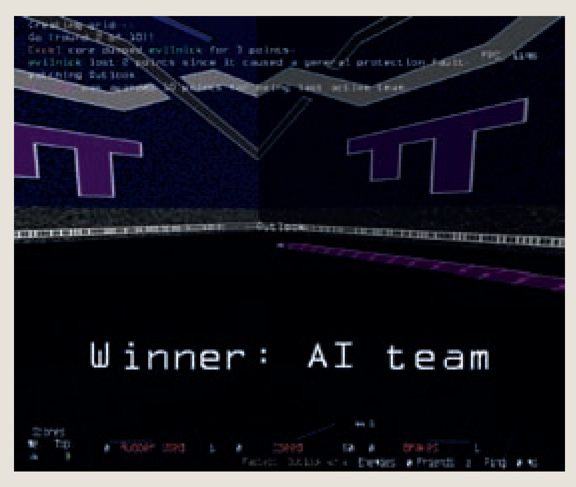
Версия 0.2.8.3.1 **Сайт** www.armagetronad.net

Наконец-то – наверное, почти намеренно подгадывая под грядущий релиз *Tron Legacy* – появилась игра, воздающая по заслугам фильму-прототипу. Запрыгивайте на световой мотоцикл и ощутите пульсацию фотонов, мчась по сетке наперегонки со своими компьютерными или онлайн-соперниками по игре *Armegatron Advanced*.

Графика довольно условная, и хотя примитивные текстуры вокруг сетки только отвлекают от ощущения схватки со злобными электронными врагами, они именно такими и должны быть (под условностью мы подразумеваем, что на самом-то деле не нарисуешь среду, которая считается состоящей из света и энергии). Сами мотоциклы и их следы удивительно напоминают фильм *Tron*, и даже если вы потерпите неудачу, врезавшись в какого-нибудь компьютерного соперника, вы упьетесь ностальгией, наблюдая, как разлетаются во всех направлениях составляющие вас фотоны. Правда, это если вы смотрели фильм; в противном случае вы будете недоумевать, зачем разработчики включили в игру столь избитый эффект.

В этой игре используется *OpenGL*, и для ее нормальной работы требуется карта с поддержкой аппаратного ускорения. Игра распространяется как файл Autopackage, так что ее установка будет проста. Меню позволит вам играть в локальной игре против компьютерных соперников (со странно знакомыми именами – «Word», «Excel» и «Outlook»; интересно, чем они вдохновили…), или можете поискать других поклонников игры в режиме онлайн, чтобы схлестнуться с ними. Есть и соответствующий серверный пакет, если вы захотите настроить машину в LAN, чтобы с ее помощью решать внутриофисные споры. Конец Строки.

> «Графика довольно условная, но такой она и должна быть.»



› Все видимое должно расти над собой и прорываться в царство невидимого.

Трубопровод

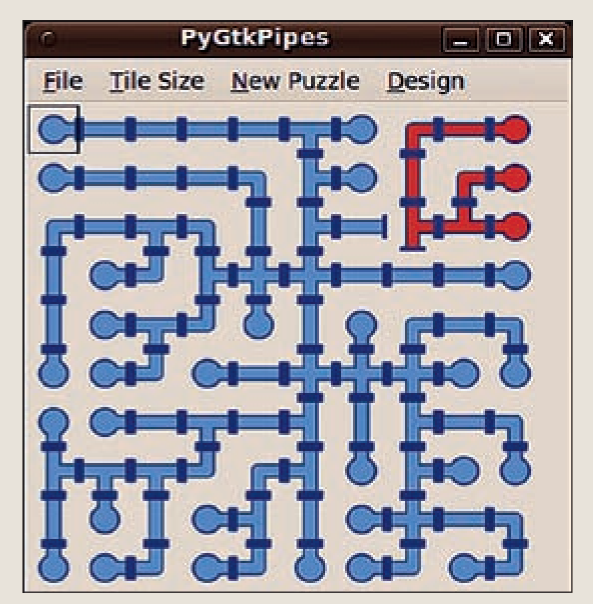
PyGTKPipes 0.3

Версия 0.3 **Сайт** http://bit.ly/aaS5iD

По неведомой причине разработчики игр во всем мире, похоже, считают сантехников недоумками. Столько игр основано на переплетении причудливо изогнутых медных труб… поневоле решишь, что средний водопроводчик слаб в геометрии. Хотя в реальности сантехники выполняют соединения недешевыми гибкими подводками из металлопластика и перекладывают оплату того, что облегчает их труд, на вас – ну и кто тут хитрее?

По любому, от подобных игр мы ждем затейливого соединения труб, и эта несложная головоломка вас не разочарует. *PyGTKPipes* – не совсем клон *Pipe Mania*: здесь нет ограничения на время и кучи неиспользованных каналов. Из некоторых труб льется вода, и придется подумать, как развернуть их элементы, чтобы вода досталась всем в решетке. Просто щелкайте по элементам – они будут вращаться по часовой стрелке и при контакте с водой окрасятся в голубой цвет. Это труднее, чем кажется, особенно на высших уровнях. Таймера нет, так что вы играете из спортивного интереса, но эта простенькая игрушка основана на Python и *PyGTK*, и не составит особого труда расширить ее, добавив свои любимые функции (скажем, счетчик ходов или возможность сохранения случайно выбранной сборки, для составления коллекции любимых труб?). Имеется меню Design; его цель, видимо, создание нового дизайна элементов, но в нашей тестовой версии меню не работало. Зависимостей в игре немного, так что, возможно, она заработает и без установки чего-то еще. Увы, инструкций тоже не в избытке; распаковав архив, для запуска игры просто введите **python pipes.py**.

> «Разработчики игр, похоже, считают сантехников недоумками.»



› «Хм-мм [трет подбородок]. Вам нужна гибкая подводка 25 мм, рублей этак за 1000…»

SSH-клиент

Pac 1.0

Версия 1.0 **Сайт** http://freshmeat.net/projects/pac

Есть вещи, которые проще и лучше делать через командную строку. Редактирование файлов настройки X, копирование больших файлов, ликвидация директории с исходниками MikeOS и, обычно, вход на серверы Telnet (добро пожаловать в 80-е) или SSH. Да, обычно. Хотя командная строка для всего этого работает быстрее, особенно если учесть, что 99,999 % времени вы будете использовать ее же на хосте, на который вы вошли, легко забывается, какая это головная боль, когда приходится постоянно обращаться к дюжине консолей маршрутизаторов или серверов.

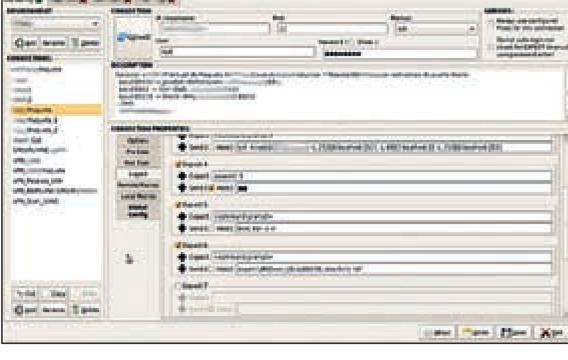
Графическая среда может отчасти облегчить все эти страдания, превратив выбор нужного вам пункта назначения из списка, запоминание мудреных паролей администратора (подскажу: «пустите-меня») и обставление всех расширенных процедур входа в простой процесс «наведи-и-щелкни».

Пока вы не замолотили по клавиатуре, набирая гневную отповедь, спешим сказать: да, есть масса причин отказаться от Telnet. Но, как это ни странно, он используется огромным количеством сетевых устройств и серверов (почтовыми серверами, коммутаторами, модемами), хотя они обычно в безопасности за брандмауэром.

Pac-спаситель

Pac сделает для вас все это в симпатичном и простом в использовании интерфейсе *GTK*. Фактически, вместе с облегченной процедурой регистрации, *Pac* использует протоколы Expect, чтобы управлять сложными согласованиями и запускать любую нужную вам команду. Представление в виде вкладок отслеживает открытые клиентские подключения, а список сохраненных соединений внизу слева облегчает задачу открытия новой сессии. Но вам все равно придется помнить, кто из хостов есть кто!

> «Telnet используется большим количеством сетевых устройств.»



› **Возможно, вы сэкономите пару миллисекунд, сохраняя свои закладки.**

Написанный на Perl, *Pac* использует *GTK* для создания интерфейса и встраивания терминалов. Последний релиз отличается улучшением интерфейса и исправлениями ошибок, но перед его запуском проверьте, не надо ли освежить ваши пакеты Perl.

Редактор текстур

NeoTextureEdit

Версия 0.5.3a **Сайт** http://bit.ly/bX7LUT

Создание чего бы то ни было отнимает время, создает мороку и – глянем правде в глаза – требует от большинства из нас таких усилий, что не стоит и браться. И если некая программа помогает что-либо создать так-легко-что-это-прямо-кайф, к ней стоит присмотреться.

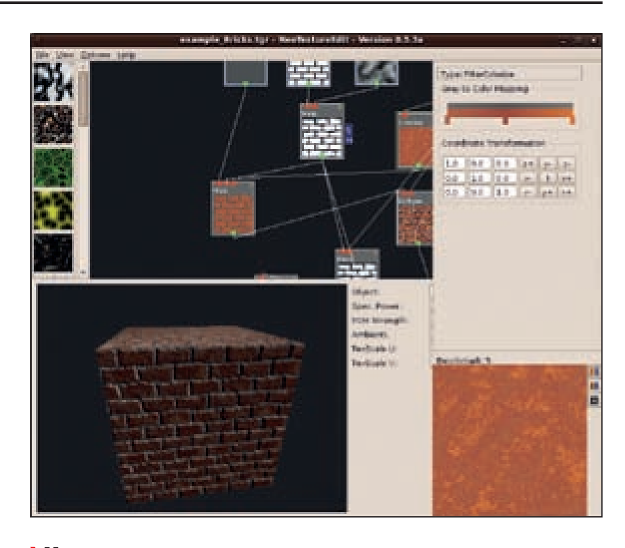
Не надо быть гением, чтобы догадаться, что исходно данное приложение создавалось для создания графических текстур – но это отнюдь не обычные текстуры.

Отличительная черта *NeoTextureEdit* – наличие области графических узлов, где можно генерировать, фильтровать, комбинировать, наслаивать и обрабатывать изображения. Линии связей показывают отношения между объектами, а ожидаемый результат отображается в миниатюре. Узлы в графической области можно передвигать, чтобы было проще видеть происходящее, а связи между ними останутся при этом нетронутыми. Текстуры здесь являются процедурными: вы для них создаете рецепт, а не большой файл с растровым изображением (хотя, конечно, можно и экспортировать PNG-графику, коли охота). На хранение процедурных текстур уходит гораздо меньше памяти – экономятся важные ресурсы для систем, требующих множества текстурных объектов.

Интерпретация текстур

Фильтров пока не в избытке, так что заменой *GIMP* в этом плане *NeoTextureEdit* не станет, но он во многом проще в использовании и понимании. Просмотр графики со связанными рабочими узлами облегчает эксперименты с различными эффектами и рецептами, и обновляющийся предпросмотр результата постоянно находится у вас перед глазами.

> «Вы создаете для текстур рецепт, а не растровое изображение.»



› **Мы гимназиев не кончали. Но школьный французский как-то раз весьма пригодился.**

Будучи кросс-платформенным, раз уж он написан на Java, *NeoTextureEdit* также выполняет достойную работу по реализации расширений OpenGL для предпросмотра части текстуры, чтобы вы смогли составить представление о том, как будет выглядеть конечный результат. Естественно, все файлы можно разложить черепицей и/или отмасштабировать.

Анализатор сети

Darkstat 3.0.713

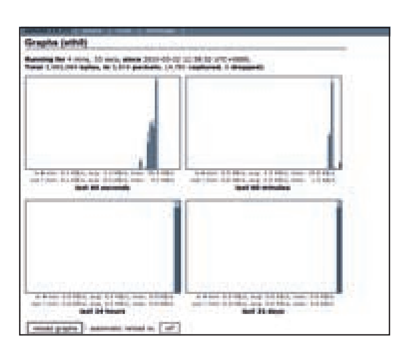
Версия 3.0.713 **Сайт** http://dmr.ath.cx/net/darkstat/

Ждете усложнить свою жизнь? Презираете легкие пути? Ну, полагаю, скоро Майк выдаст новый релиз MikeOS. А всем остальным стоит обратить внимание на программу учета сетевой статистики *Darkstat*.

У нас для этой цели уже есть *ntop*, и он отлично показывает, что происходит в данный момент, но если вы не можете постоянно следить за столбцами и строками цифр, ползущими по экрану, то для долгосрочного наблюдения он не вполне годится. *Darkstat* не претендует на гениальность, он просто отчитывается по сетевой статистике, зато предоставляет скользящее среднее, таблицы и протокольные отчеты.

Если его хорошенько попросить, он сделает обратные DNS-запросы, отобразит список портов, создаст графики – и все это на крошечном web-сервере, почти не отнимающем циклов процессора.

Для работы *Darkstat* требует немногим больше, чем *libcap* (который у вас, скорее всего, уже есть), и прекрасно компилируется обычным способом. Можно настроить его позаковыристее, но, вероятно, вы сочтете конфигурацию по умолчанию вполне приемлемой. Запустите приложение с параметрами **-i eth0**, чтобы начать слежение за основным Ethernet-интерфейсом, и вперед! По умолчанию вы сможете просматривать страницы web-интерфейса на **http://localhost:667**, но можете изменить и порт тоже, если захотите (для этого используется ключ **-p**).



› *Darkstat* снабжен читаемыми man-страницами на случай затруднений, хотя этим пренебрегают многие полезные инструменты.

Программа для печати этикеток

gLabels 2.2.7

Версия 2.2.7 **Сайт** http://glabels.org

Вещам нужны этикетки. Иначе как узнать, что это такое? И – стоп! Правильно ли все расставлено у вас в шкафу? Господи, надо немедля наклеить этикетки на все подряд!

Для изготовления этикеток к банкам с крупой или домашним вином можно использовать и какой-нибудь *OpenOffice.org*, *Inkscape* или *Scribus*, но, честно говоря, пока вы забьете туда свой текст (или, в случае с *OpenOffice.org*, пройдете полпути до его загрузки), *gLabels* уже успеет напечатать пяток страниц.

Да, *gLabels* не лишен специфики, но работает очень хорошо. Главный ключ к успеху здесь в том, что он знает все о канцелярских наборах перфорированных визиток или листов для наклеек и поможет вам напечатать их, не тратя первые три страницы на разметку изображений и текстовой части. Вы можете запросто напечатать несколько сотен карточек или ярлыков или использовать функцию простановки почтовых реквизитов, чтобы создать набор ярлыков/карточек в одном стиле. Программа естественным образом поддерживает любые шрифты, имеющиеся в вашей системе, и вы можете загружать растровую графику или файлы SVG, чтобы оживить свое творчество. Здесь даже имеется встроенный генератор штрих-кодов, если вы захотите пометить им свои коллекции или добавить данные для машинного считывания. LXF

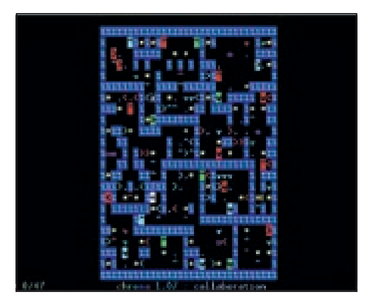


› *gLabels* воплощает дух Unix: делать что-то одно, но зато хорошо.

Также вышли

Новые и обновленные программы, тоже достойные внимания…

» Box 0.1.3
Язык, нацеленный на завоевание мира векторной графики.
http://sourceforge.net/projects/boxc



› Если у вас еще не болит голова, стукнитесь ею об стол.

» Chroma 1.11
Психоделическая игра – поиск парных изображений.
www.level7.org.uk/chroma

» Eric4 4.4.3
Лучшая (?) IDE Python потихоньку берется за поддержку Python 3.
http://eric-ide.python-projects.org/index.html

» Parted Magic 4.9
Всесторонний дистрибутив для восстановления/форматирования диска.
http://partedmagic.com

» Ocre 0.034
Программа оптического распознавания символов в ASCII или Unicode.
http://lem.eui.upm.es/ocre.html

» SoundConverter 1.5.0
В соответствии с названием – превратите свой FLAC в OGG и т. п.
http://soundconverter.berlios.de

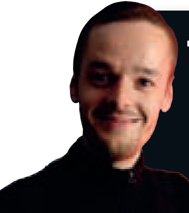
» USBManager 1.0
Аккуратная утилита для работы с портативными USB-устройствами хранения данных.
https://launchpad.net/usbmanager

» Viking 0.9.91
Создавайте, редактируйте, просматривайте и конвертируйте маршруты GPS с исключительной точностью.
http://bit.ly/dltG6S

» Wiki on a Stick 0.11.0
Самая популярная в мире реализация персональной wiki обрела лучший интерфейс.
http://stickwiki.sourceforge.net

На диске

Все, что вам нужно, чтобы скоротать время до выхода LXF133…



Читайте, чтобы найти всю информацию, которая вам понадобится для использования этого DVD! Если вы новичок в Linux, откройте index.html на диске (Сторона 1) и перейдите в раздел Справка: там вы найдете мини-учебник, дающий ответы на вопросы справа.

Майк Сондерс, редактор диска
mike.saunders@futurenet.com

- » Что такое Linux?
- » Что такое дистрибутив?
- » Загрузка ПК с DVD
- » Разбиение жесткого диска на разделы
- » Навигация по файловой системе
- » Использование командной строки
- » Установка программ

Свободное искусство

Open Clip Art 2.0

Когда при работе над каким-то документом вам понадобятся ил= люстрации к заданной теме – например, компьютеры, здания или животные – вы, наверное, обратитесь за ними к Google. И, наверное, вы будете продираться через заросли сайтов, якобы предлагающих бесплатный клипарт, но на деле принуждающих подписаться на всякие рассылки просто за доступ к файлам. А потом, наверное, окажется, что эти файлы – дрянные растровые изображения с низким разрешением, и на них красуется водяной знак – URL сайта.

Open Clip Art отметает такие проблемы. Он не накладывает ограничений на то, что можно делать с картинками из его коллекции – и никаких отчислений автору, никакого максимального числа использований, ничего подобного. Изображения – общественное достояние и могут применяться где угодно и столько раз, сколько вам нужно.

Open Clip Art 2.0 (в разделе **Рабочий стол**) содержит более 13000 изображений в формате SVG (векторном), и вы сможете открыть их в редакторе типа *Inkscape* и сгенерировать растровое изображение любого нужного размера без потерь качества. Скопируйте **openclipart-2.0-full.zip** в домашнюю директорию и распакуйте его, используя свой менеджер файлов. Если в последнем есть функция предпросмотра SVG-изображений, как, скажем, в *Nautilus*, можете перейти в директорию **clipart** и затем переключаться по поддиректориям, чтобы просматривать изображения.

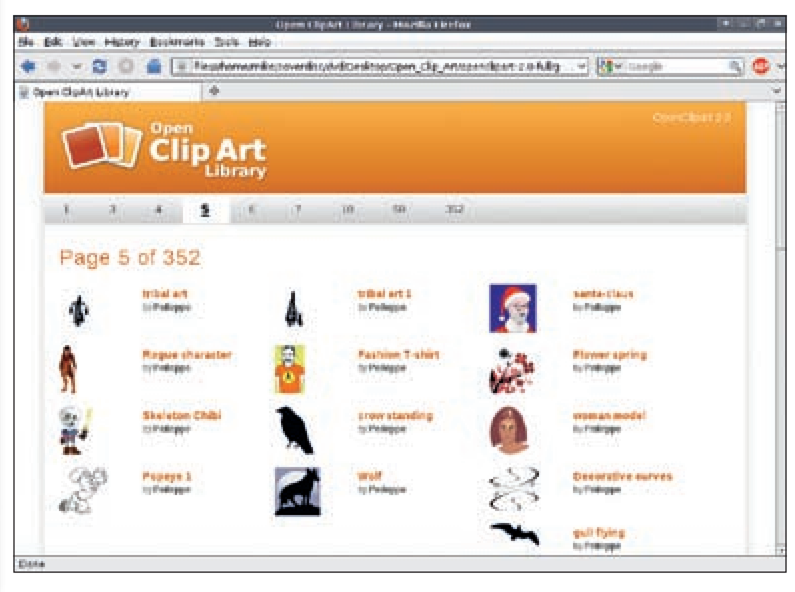
Еще один способ обозревать архив – HTML-интерфейс. Зайдите в директорию **gallery** и откройте **index.html** для навигации по страницам, содержащим миниатюры клипарта. (Иногда попадаются имена файлов с автором «undefined» – «не определен» – и без миниатюры; это из-за ошибки в скрипте, генерирующем в Open Clip Art галерею, так что просто игнорируйте их.)

Для нахождения изображения по заданной теме можно обратиться к функции поиска вашего файлового менеджера. Или – воспользуйтесь командной строкой, чтобы определить местонахождение файлов и открыть их автоматически. Например, если вы распаковали **openclipart-2.0-full.zip** в свою домашнюю директорию, откройте терминал и введите:

```
cd openclipart-2.0-full/clipart/
```

Теперь вы перешли в директорию **clipart**. Чтобы найти все файлы, содержащие слово «kitten» («котенок») (неважно, в каком регистре), введите

```
find . | grep -i kitten
```



› **Просмотр обширной коллекции Open Clip Art через HTML-интерфейс.**



Важно ВНИМАНИЕ!

» Перед тем, как вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим.

Диски *Linux Format* DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция *Linux Format* не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензии.

Бракованные диски

В маловероятном случае обнаружения бракованного диска *Linux Format*, просим связаться с нашей группой поддержки по адресу disks@linuxformat.ru для получения содействия.

Вы увидите список SVG-файлов (и их PNG-миниатюр), содержащих «котенка». Чтобы открыть SVG файлы в *Inkscape*, припишите к команде следующий фрагмент:

```
find . | grep -i kitten | grep svg | xargs
inkscape
```

Программы из Обзоров
Amarok и Bluefish

Как вы, возможно, уже увидели в нашем разделе Обзоров, мы вывели на сцену новые версии *Amarok* и *Bluefish*. Исходный код для них вы найдете на DVD, в разделах **Рабочий стол** и **Интернет** соответственно. *Amarok* применяет для сборки *CMake*, и ему нужны библиотеки разработчика KDE 4 и предварительно установленные заголовочные файлы. Перейдите командой **cd** в директорию, созданную после того, как вы распаковали исходный код, и введите:

```
cmake -DCMAKE_INSTALL_
PREFIX=`kde4-config --prefix`
make
sudo make install
```

А у *Bluefish* имеются пакеты для Fedora и Ubuntu – для его установки дважды щелкните по файлу, соответствующему вашему дистрибутиву. В других дистрибутивах придется компилировать *Bluefish* из исходного кода, как обычно, скриптом *configure*. Распакуйте исходный код, перейдите (**cd**) в получившуюся директорию и введите:

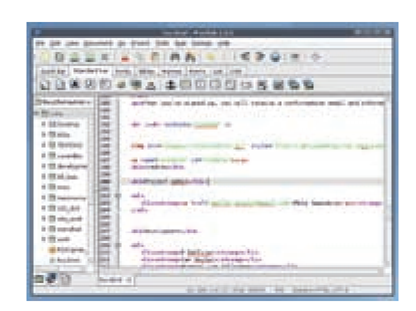
```
./configure
make
sudo make install
```

После установки, запускайте приложение вводом **bluefish** или добавьте иконку запуска на рабочий стол или в менеджер окон.

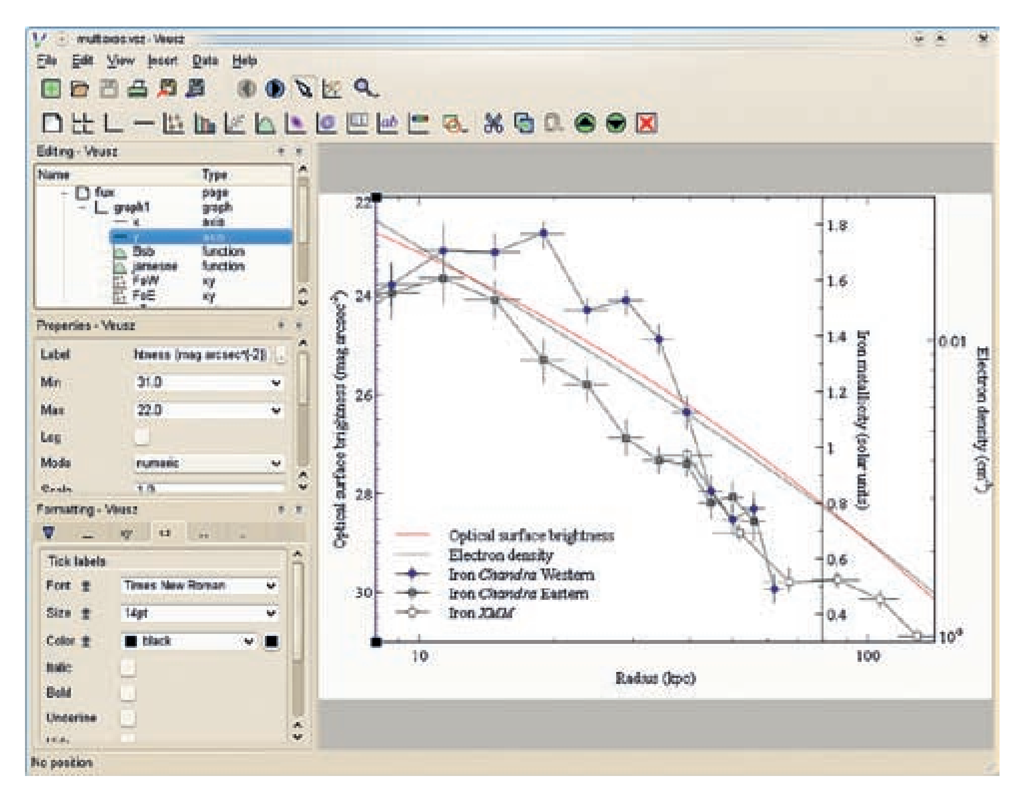
Ну, а если все это кажется вам китайской грамотой, зайдите в раздел **Справка/Новичку в Linux** на DVD и прочтите краткое руководство по использованию командной строки и установке программ – и вы получите представление о том, что происходит.

Другие программы
CherryTree, Banshee, Veusz...

Если вы ищете приложение для создания заметок, где было бы больше функций, чем в обычных желтых стикерах, поставляемых с большинством дистрибутивов, попробуйте *CherryTree* из раздела **Рабочий стол**. Оно хранит заметки в иерархической структуре, что позволяет иметь записи, проистекающие из других заметок; оно также поддерживает форматирование с возможностью вставки изображений. Вместе с исходным кодом вы найдете Deb-файл, который можно установить на Ubuntu, Debian и других дистрибутивах на базе Debian.



› **Bluefish 2.0 доступен в виде пакетов для Fedora и Ubuntu, и можно скомпилировать исходный код самостоятельно.**



› **Veusz — жутко мощный инструмент для создания графиков, с огромным количеством опций.**

А вот недавний релиз *Banshee 1.6* подарил немало новых функций этому музыкальному плейеру. «Auto DJ» постоянно заботится о пополнении вашего плейлиста, и вы можете просматривать альбомы в виде сетки и импортировать песни из www.eMusic.com. Чтобы сделать это приложение еще более разносторонним, реализован механизм расширений – продвинутые пользователи могут создавать собственные мини-приложения для работы за пределами *Banshee*: например, поиск слов к песням и инструменты потокового Интернет-радио.

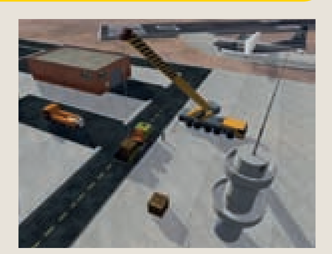
Читателям с научным складом ума мы припасли *Veusz*, пакет для изготовления графиков, позволяющий создавать «готовые к публикации графики, графики XY и гистограммы». Вы можете работать в нем как через GUI, так и через командную строку; можно также задействовать скрипты. Для экспорта построенных вами графиков здесь предусмотрены форматы PostScript и PDF.

И это еще не все!
Пять превосходных игр

FlightGear, как говорилось на стр. 11, был одной из самых многообещающих «развлекательных» программ с открытым кодом с самого начала разработки в 1997 (хотя многие назвали бы его очень серьезным приложением). На DVD вы найдете **FlightGear-2.0.0.tar.gz** – файл размером 2,4 МБ, содержащий основной движок для симулятора и архив с данными для самолетов и ландшафта. Летные симуляторы отличаются сложностью в управлении, так что перед первым взлетом рекомендуем прочитать документацию в **getstart.pdf** (на английском языке).

Среди более легких игр, *Brain Party* является собой забавную компиляцию головоломок – ее написал Пол Хадсон, чей портрет вы можете посозерцать на стр. 88. *Brain Party* состоит из 36 мини-игр, которые проверят вашу логику, математические знания и реакцию и высчитают вес вашего мозга на основе полученных результатов. Распакуйте архив и запустите для компиляции **make**, а затем **./brainparty**, чтобы запустить ее.

В сфере 3D у нас есть *Rigs of Rods*, физический симулятор, в котором задействованы грузовики, краны, самолеты и корабли. Его уникальность заключается в использовании физики мягких тел, поэтому объекты в нем обладают определенной гибкостью, приводящей к весьма реалистичным коллизиям. И еще у нас имеется



› ***Rigs of Rods* провоцирует вас сталкивать автомобили. Это круто.**

Stendhal, весьма впечатляюще сделанная онлайн-ролевка, и новый релиз *SuperTux* – всеми любимой платформенной игры, почерпнувшей вдохновение в небезызвестной *Mario*.

На диске

Четыре классных дистрибутива Linux, работающих прямо с DVD

Дистрибутив Linux

Дистро-революция

LXFDVD этого месяца – с четверной загрузкой, то есть мы создали особый диск, способный загрузить для вас напрямую четыре уникальных дистрибутива Linux. Просто запустите компьютер с DVD в приводе – и увидите загрузочное меню, позволяющее выбрать дистрибутив Linux клавишами курсора и нажатием Enter. Если ваш компьютер не загружается с DVD, возможно, надо изменить настройки BIOS: зайдите в BIOS во время загрузки компьютера (обычно для этого нужно нажать Esc, Del или F2), поищите опцию порядка загрузки и проверьте, чтобы загрузка с DVD стояла до загрузки с жесткого диска.

Вы можете прочитать все о прекрасных функциях, предлагаемых этими дистрибутивами, в материале на стр. 22; здесь мы просто дадим вам несколько подсказок по работе с ними.



› Запустите компьютер, вставив в дисковод LXFDVD, и выберите свой новый дистрибутив.

» Dreamlinux 3.5

Первым пунктом в меню загрузки идет Dreamlinux. Это *Xfce*-версия дистрибутива (вместо версии с Gnome по умолчанию), и она работает в режиме Live, так что вы сможете познакомиться с ней, не устанавливая на жесткий диск. Если при загрузке не возникнет никаких проблем, через несколько минут перед вами появится рабочий стол. Чтобы запустить программы, щелкайте по значкам внизу экрана или воспользуйтесь пиктограммой вверху слева, чтобы открыть большое меню. Закончив работу, вы можете щелкнуть по тому же значку и выбрать Quit, чтобы выйти из системы.

» Slitaz 3.0

Далее идет Slitaz – легковесный Live-дистрибутив, идеальный для старых машин. Когда он загрузится, появится текстовое меню, предлагающее выбрать язык и раскладку клавиатуры: для этого используйте клавиши курсора (или кнопки Home/End, чтобы перепрыгивать вниз-вверх по спискам) и Enter. Затем загрузится рабочий стол. Это минималистская среда, но вы можете просматривать программы, щелкая по темно-красному значку в нижнем левом углу.

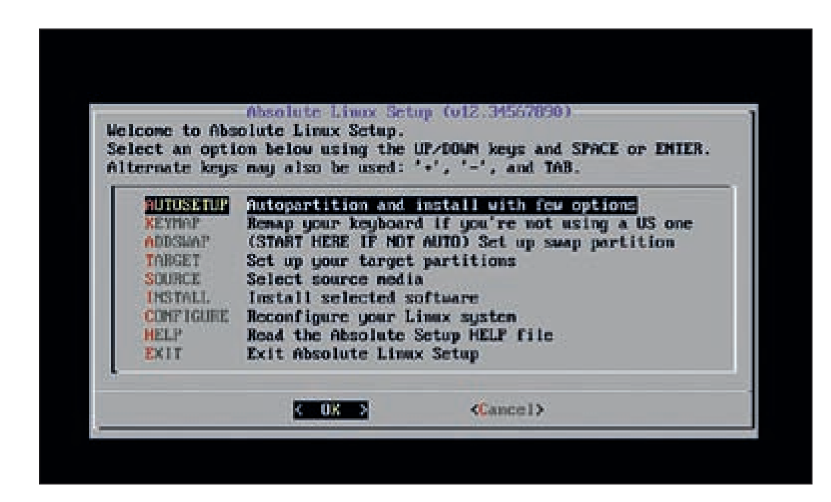
Учтите, что некоторые инструменты администрирования запускаются только от имени root (суперпользователя), так что вам будет предложено ввести пароль. Пароль – просто «root», и большинство диалоговых окон напомнят вам об этом, когда вы будете использовать этот дистрибутив. Чтобы выйти из системы, щелкните по красной кнопке и выберите Logout; перед вами появится диалоговое окно, которое спросит вас, хотите ли вы выключить компьютер или перезагрузить его.

Хотя Slitaz загружается прямо с DVD, мы записали на диск также его ISO-образ, специально для машин-ветеранов. Зайдите в раздел **Дистрибутивы/Slitaz**, и вы увидите файл **.iso** – можете прожечь его на CD-R и загрузиться с него на машине, не имеющей DVD-привода. Помните, что вам надо записать его как ISO-образ, а не просто скопировать файл на CD-R; если вы не уверены в том, как это делается, загляните в документацию вашей программы по прожигу CD. В командной строке введите

```
cdrecord -v -dao dev=/dev/cdrom slitaz-3.0.iso
```

» Absolute Linux 13.1.0

Предыдущие дистрибутивы работают в режиме Live, а вот Absolute нужно сначала установить. Этот процесс происходит



› Программа установки Absolute Linux работает в текстовом режиме, но пользоваться ею не сложно.



› Slitaz, стартуя, задаст вам пару вопросов о вашем местоположении, прежде чем выкатить рабочий стол.

в текстовом режиме, посредством инсталлятора, заимствованного из Slackware, и хотя он и не так дружелюбен к новичкам, как графические программы установки других дистрибутивов, но все же не слишком сложен. После загрузки появится меню: вы можете выбрать индивидуальную раскладку клавиатуры, настроить раздел подкачки и т. п., или выбрать опцию Autosetup, чтобы выполнилась экспресс-установка. Но тогда Absolute Linux захватит весь диск целиком, так что выбирайте эту опцию, только предварительно создав резервные копии важных данных!

По завершении установки система перезагрузится. Извлеките DVD из дисковода (чтобы установка не повторилась по новой), и Absolute Linux запустится уже с жесткого диска.

» Unity Linux 2010 RC1

И, наконец, у нас есть Unity. Как вы увидите в меню загрузки, для этого дистрибутива имеется масса опций. В большинстве случаев наилучшим выбором для запуска системы в режиме Live будет первая опция, «Unity Live start», но вы также можете выбрать один из безопасных вариантов («safe»), если при загрузке возникнут проблемы (например, если ваша видеокарта не определилась правильно). Если у вас ОЗУ на 1 ГБ или больше, попробуйте опцию «Copy to RAM» («копировать в ОЗУ») – это значительно увеличит скорость дистрибутива, поскольку его не придется считывать с DVD.

После запуска Unity предложит вам выбрать язык и раскладку клавиатуры, и появится окно приглашения. Вы можете войти в систему под именем «guest»; а если вдруг потребуются действия от имени root (администратора), то пароль его и есть «root». Напоминание об этом вы увидите в верхнем левом углу экрана приглашения.

Unity содержит не особенно много готовых программ, и вы можете просмотреть включенные в него приложения через значок с зеленым листиком в нижней левой части экрана. Чтобы завершить работу, дважды щелкните по значку Halt на рабочем столе. Помните, что машина выключится сразу, безо всяких напоминаний о необходимости сохранить плоды ваших трудов!

В теме этого номера обсуждается использование ISO-образов с программами виртуализации. Сейчас в вашем распоряжении имеется физический диск, который вы получили со своим экземпляром Linux Format, но если вам требуется превратить его в ISO-образ, это можно сделать одной-единственной командой. Вставьте **LXF**DVD в дисковод, откройте терминал и введите

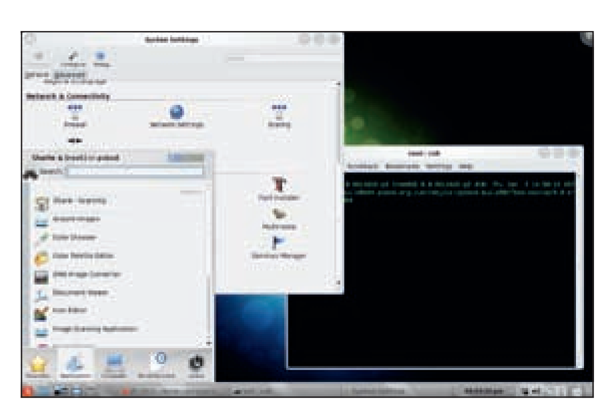
```
dd if=/dev/dvd of=lxfdvd132.iso
```

Выполнение команды займет несколько минут, но в итоге у вас появится файл **lxfdvd132.iso** – точный ISO-образ физического диска. И вы можете использовать его в программах виртуализации.

Вторая сторона
PC BSD 8

О том, что представляет собой дружественная к пользователю операционная система PC BSD 8, подробно рассказывается на стр. 9. Мы не будем здесь лишний раз повторяться, и просто покажем, как установить ее на жесткий диск.

Для начала, загрузите компьютер со второй стороны **LXF**DVD. Рекомендуется CPU Pentium 4 и выше, 512 МБ памяти и 12 ГБ на жестком диске. Спустя непродолжительное время перед вами появится графический установщик системы (если этого не произойдет, попробуйте перезагрузить компьютер и выбрать в загрузочном меню вариант 2 – ACPI Disabled). Укажите язык и раскладку клавиатуры, а затем определитесь, что вы будете устанавливать: PC BSD или оригинал, FreeBSD. Затем инсталлятор предложит разбить диск (выберите опцию Custom, если намереваетесь задействовать файловую систему ZFS), задать пароль root и обычного пользователя, и выбрать компоненты (web-браузер, почтовый клиент, исходные тексты и т. п.), подлежащие установке. Система приступит к копированию файлов, после чего следует извлечь диск из привода и перезагрузиться. **LXF**



› PC BSD позиционируется как дружественная к пользователю настольная система.

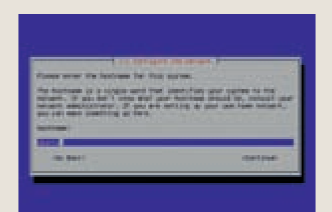
Межсетевой экран eBox 1.4

Со стр. 16 начинается наше Сравнение брандмауэров – т.е. дистрибутивов Linux, разработанных, чтобы превратить ваш компьютер в сетевое устройство и пресечь попытки вторжения в ваши машины из Интернета. На диск записан победитель Сравнения, eBox – он находится в разделе **Дистрибутивы/eBox**. Это ISO-образ, и для установки его нужно прожечь на CD-R. Используйте свою программу прожига CD, чтобы записать **ebox_installer-14-2.iso** на диск непосредственно как образ диска; его нельзя просто скопировать, как обычный файл. Если вы не уверены в том, как это сделать, подробности вы найдете в документации вашей программы для прожига дисков. Кроме того, можно записать его из командной строки таким образом:

```
cdrecord -v -dao dev=/dev/cdrom
ebox_installer-14-2.iso
```

Если вам когда-либо случалось устанавливать Ubuntu через его вариацию Alternate, то вы уже знакомы с программой установки eBox – она та же самая. Она работает в текстовом режиме; для перехода от одного элемента экрана к другому служит клавиша Tab, а кнопки выбираются нажатием на Enter. В варианте загрузки по умолчанию eBox не позволяет выполнять продвинутые действия по делению диска на разделы – например, создать множественную загрузку с другими дистрибутивами или операционными системами; он просто перепишет весь диск целиком.

Машине, служащей сетевым брандмауэром, множественная загрузка, возможно, и ни к чему; но все же об этом стоит помнить. Более подробную информацию по eBox вы найдете в документе **Network-Administrators.pdf** (на английском языке), который также находится в разделе **Дистрибутивы/eBox** на диске. Это очень подробное руководство на 230 страницах, охватывающее вопросы установки, использования и администрирования. Оно объясняет, как настроить сеть, создать резервную копию системы и установить обновления безопасности.



› Победитель нашего Сравнения, eBox, превратит бесхозный компьютер в мощный брандмауэр.

Информация о диске

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Внимательно прочтите это перед тем, как использовать LXFDVD!

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

» **имя_программы-1.0.1.i386.rpm** – вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
» **имя_программы-1.0.1.i386.deb** – такой же пакет, но уже для Debian;
» **имя_программы-1.0.1.tar.gz** – обычно это исходный код;
» **имя_программы-1.0.1.tgz** – тот же файл, что и выше по списку: «tgz» – это сокращение от «tar.gz»;
» **имя_программы-1.0.1.tar.bz2** – тот же файл, но сжатый bzip2 вместо обычного gzip;
» **имя_программы-1.0.1.src.rpm** – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
» **имя_программы-1.0.1.i386.FC4.RPM** – двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
» **имя_программы-1.0.1.ppc.Suse9.rpm** – двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
» **имя_программы-devel-1.0.1.i386.rpm** – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Июнь 2010 LXFDVD 132

LINUX FORMAT

PC-BSD 8

Ubuntu мира FreeBSD: прост в установке, прост в использовании, прост в управлении

» Графический инсталлятор с поддержкой файловой системы ZFS
» Рабочий стол KDE 4 и проприетарные драйверы NVIDIA
» Установка пакетов в формате PBI одним щелчком мыши

А ТАКЖЕ: дополнительные модели для FlightGear

Июнь 2010 LXFDVD 132

LINUX FORMAT

Дистро революция!

Четыре интересных дистрибутива, упомянутых в теме номера, ждут установки в вашу виртуальную машину

» **OpenClipart** 13423 векторных изображения, доступных по свободной лицензии
» **FlightGear 2.0** Взмывайте к небесам в этом реалистичном авиасимуляторе
» **eBox 1.4** Победитель нашего Сравнения готов защитить вашу сеть

А ТАКЖЕ: средства разработки, настольное ПО и игры!

Содержание

LINUX FORMAT

Сторона 1

Сторона 2

Пожалуйста, перед использованием данного диска ознакомьтесь с опубликованной в журнале инструкцией!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь пожалуйста по адресу disks@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не может принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Тираж изготовлен ООО «Уральский электронный завод». 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПТР ВАФ 77-15.



Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что если у вас нет устройства, с помощью которого можно было бы записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков: подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика.

Школа LXF



Обмен опытом и передовые идеи по использованию свободного ПО в образовании

Свободные ПО и электроника



Что общего у среды программирования для детей *Scratch* и свободной вычислительной платформы *Arduino*? **Александр Казанцев** даст ответ, сделав управление аппаратными устройствами доступным школьнику.



Наш эксперт

Александр Казанцев
К. т. н., доцент кафедры информатики Глазовского государственного пединститута, руководитель проекта EduMandriva, автор локализаций (и разработок) для Mandriva, *Klavaro*, *Gambas2* и *LXDE*.

Программирование – увлекательная вещь, но результат своего труда можно увидеть лишь на экране. То, что можно потрогать физически, вызывает больший интерес, чем виртуальная копия. Изучение алгоритмов дает понятие «исполнителя», что в большинстве детских умов ассоциируется с роботами или хотя бы электронными устройствами. Программируя реальное оборудование и взаимодействуя с ним, учащийся лучше понимает, для чего в принципе нужна программа и как работают вычислительные системы в целом. Наконец, когда какая-то «железка» вдруг выполняет твои команды, или персонаж на экране начинает слушаться только что собранной электронной схемы – это просто интересно и познавательно.

И здесь нам помогут свободные ПО и электроника. Проект Arduino (см. врезку справа) позволит войти в этот мир с минимальными затратами. Единственное «но» – программы (т. н. «скетчи») для используемых в Arduino микропроцессоров (ATMega) пишутся на ассемблере или с использованием специальных трансляторов с других языков. Это уровень студентов вуза, причем продвинутого. Среда *Arduino IDE* представляет верх минимализма и аскетизма и требует знания языков уровня С или Java. Как применять это в школе?

Спешу вас успокоить – мы будем использовать плату Arduino вместе со *Scratch* (см. врезку внизу). *Scratch* – это среда программирования «из кирпичиков», основанная на Logo, и с ней может справиться ребенок, начиная с 10–12 лет (при эмуляции ScratchBoard) или 14–15 лет – в случае *Catenary*.

Исходные материалы

Итак, нам понадобятся: среда программирования *Scratch* версии не ниже 1.3 (мы возьмем 1.4.0), плата *Arduino* (подойдет любая совместимая; у нас была Arduino Duemilanove), светодиоды трех цветов, кнопка, резисторы на 1 кОм (есть в любом магазине радиотоваров) и провода. Не помешают макетная плата или паяльник, хотя можно собрать все, просто скрутив детали (конечно, это не так красиво и надежно). Разумеется, нужен дистрибутив Linux. Вы можете повторить эти действия и под Windows, но разбираться придется самим.

Первым делом установим *Scratch*. Для этого воспользуйтесь менеджером пакетов вашего дистрибутива или скачайте последнюю версию с сайта проекта (правда, найти ее там непросто). Установка и настройка ПО для Arduino освещены в номерах **LXF**, упомянутых во врезке, и этого мы касаться не будем. Не забудьте приготовить саму плату и будьте осторожны при обращении с электроникой – не касайтесь ее голыми руками. Вам это не повредит, но детали можно «убить» статическими электричеством.

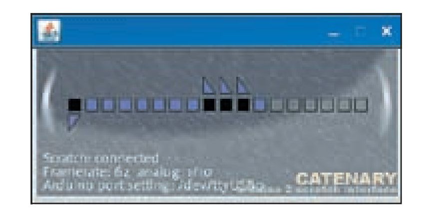
Управляем из Scratch

Мы не станем заниматься «настоящим» программированием – взамен, будем управлять платой Arduino напрямую из *Scratch*, не загружая в нее скетчи. И в этом нам поможет проект *Catenary* (http://scratchconnections.wik.is/User:Chalkmarrow/Catenary).Распакуйте архив с программой в вашу домашнюю директорию и действуйте по приведенной инструкции.

Сперва подключите Arduino к компьютеру и залейте в нее код, превращающий ее в плату ввода/вывода. Запустите *Arduino IDE* (обычно командой **arduino**) и убедитесь, что в меню Tools > Board и Tools > SerialPort выбраны правильная плата и порт. Далее, зайдите в File > Examples > Firmata и выберите StandartFirmata. После того, как скетч откроется, залейте его на плату с помощью File > Upload I/O или кнопки на панели меню. Плата готова к управлению. Не забудьте подключить светодиод к цифровому контакту 13 (одну ножку в GND, другую в DIGITAL 13).



› **Не забудьте подключить сенсоры!**



› Catenary отображает пользователю информацию в таком «минималистичном» виде.

Запустите *Scratch*, откройте в нем проект, идущий в архиве с программой – **Scratch_Catenary1(Blink).sb**, и (ни в коем случае не запуская его) включите соединение с удаленным сенсором, выбрав синюю вкладку Сенсоры и кликнув правой кнопкой на любом блоке со словом «сенсор».

Затем запустите *Catenary* (желательно от имени root, чтобы не иметь проблем с доступом к устройствам, и обязательно после *Scratch*). Перейдите в директорию **Catenary/application.linux/** и скомандуйте:

```
chmod +x ./Catenary
sudo ./Catenary
```

В моей системе (EduMandriva) *sudo* не настроен, и нужно использовать *gksu* или *kdesu*.

```
gksu ./Catenary
```

Если ваша система – 64-битная, *Catenary* откажется работать, сославшись на 32-битные библиотеки. Чтобы исправить это, создайте символьную ссылку на 64-битную библиотеку **librxtxSerial.so** из **/usr/lib64/rxtx** или **/usr/lib64/rxtx-x.y.z**. Находясь в одном каталоге с запускаемым файлом, наберите (**x.y.z** нужно заменить на актуальную версию):

```
rm ./librxtxSerial.so
ln -s /usr/lib64/rxtx/librxtxSerial.so.x.y.z ./librxtxSerial.so
```

Кроме того, нужно скомандовать:

```
rm ./lib/RXTXcomm.jar
ln -s /usr/share/java/RXTXcomm.jar ./lib/RXTXcomm.jar
```

В процессе написания статьи я обнаружил, что данные передавались на плату, но не влияли на состояние выходов. Несколько часов поисков показали, что дело еще в одной библиотеке. Возьмите файл **processing-arduino-0017.zip** с LXFDVD и замените библиотеку в **/Catenary/application.linux/lib** на содержащуюся в архиве (**/arduino/library/Arduino.jar**), не забыв назвать ее с маленькой буквы. Это может и не потребоваться; но предупрежден – значит, вооружен.

Наконец, появится окно, изображенное выше. Первый слева голубой квадрат – цифровой контакт 2, последний – 13. Серыми квадратиками обозначены аналоговые контакты 0–5 (в нумерации *Scratch* – 14–19).

Теперь вернитесь к *Scratch* и попробуйте запустить проект. Если все пойдет нормально, вы увидите мигающий светодиод.

Разберем, как устроена программа. Второй персонаж (Catenary) нужен для обеспечения работы переменных, и трогать его не нужно. Программа пишется в спрайте «кота». Все передаваемые команды начинаются с **^**. Сперва идут блоки инициализации платы – задания порта (**^arduinoPort**) и сброса состояния (**^reset**).

```
передать ^arduinoPort 0
ждать 0.5 секунд
передать ^reset
ждать 1 секунд
```

Далее мы переключаем цифровой контакт 13 на вывод (**^pinMode 13 output**).

```
передать ^pinMode 13 output
```

Подробнее о Scratch

Если вы ранее не сталкивались со *Scratch*, вам стоит познакомиться с ним поближе. Это новая среда программирования, которая позволяет детям создавать собственные анимированные и интерактивные истории, игры и другие произведения. *Scratch* базируется на традициях языка Лого и написан на Squeak. В среде *Scratch* используется метафора кирпичиков Лего, собирать из которых простейшие конструкции могут даже самые маленькие дети. Но, начав с малого, можно развивать и расширять свое умение строить и программировать. *Scratch* создавался специально для того, чтобы подростки 10–16 лет использовали его самостоятельно в сети внешкольного обучения.

На самом деле это больше, чем среда программирования – это инструмент создания интерактивных мультфильмов, музыки, игр, историй, которыми можно поделиться с другими. Дети могут завести себе учетную запись на сайте проекта http://scratch.mit.edu/ и получать/давать доступ к своим проектам из любой точки земного шара. Наличие специального модуля позволяет встраивать проекты *Scratch* в любой web-сайт (требуется Java), поэтому он может в какой-то мере служить заменой Flash. Вы можете найти больше информации по *Scratch* на таких сайтах, как http://scratched.media.mit.edu, http://letopisi.ru, http://ru-scarystories.blogspot.com.

После этого начинаем мигать светодиодом, переводя состояние цифрового канала из выключенного (**low**) в включенное (**high**).

```
всегда
  ждать 1 секунд
  передать ^digitalWrite 13 high
  ждать 1 секунд
  передать ^digitalWrite 13 low
```

Ниже приведена модификация программы, передающая азбукой Морзе сигнал SOS (три коротких, три длинных, три коротких). Чтобы добавить новый текст в блок Передать, щелкните на стрелочке сбоку, выберите Новый, и введите нужный текст.

```
когда щелкнут по [флажок]
передать ^arduinoPort 0
ждать 0.5 секунд
передать ^reset
ждать 1 секунд
передать ^pinMode 13 output
повторить 3
  передать ^digitalWrite 13 high
  ждать 0.5 секунд
  передать ^digitalWrite 13 low
  ждать 0.5 секунд
ждать 0.5 секунд
повторить 3
  передать ^digitalWrite 13 high
  ждать 1 секунд
  передать ^digitalWrite 13 low
  ждать 1 секунд
повторить 3
  передать ^digitalWrite 13 high
  ждать 0.5 секунд
  передать ^digitalWrite 13 low
  ждать 0.5 секунд
```

Скорая помощь

После нажатия кнопки сброса на плате и других подобных действий нужно действовать по алгоритму:
1. Подключить Arduino и залить прошивку.
2. Запустить *Scratch*, включить связь с сенсорами и написать программу.
3. Запустить *Catenary*.
4. Запустить программу в *Scratch*.

»

Arduino и его родственники

LXF уделял Arduino немало внимания (см. номера 100–101 и 103–106 на wiki.linuxformat.ru), поэтому мы остановимся на том, где найти эти устройства в России и как получить помощь на родном языке. Оригинальные платы Arduino появляются (по мере поступления) в интернет-магазине ГНУ/Линуксцентра (www.linuxcenter.ru). Существуют два других проекта, выпускающих свои Arduino-совместимые платы: Robocraft (http://robocraft.ru/) и Freeduino (http://freeduino.ru/). На их сайтах проектов можно найти много полезной информации по работе с платой.

Не только выход

Но *Catenary* позволяет не только управлять цифровыми выходами – вы также можете изменять состояние аналоговых контактов и получать информацию с цифровых и аналоговых входов, то есть подключать внешние датчики.

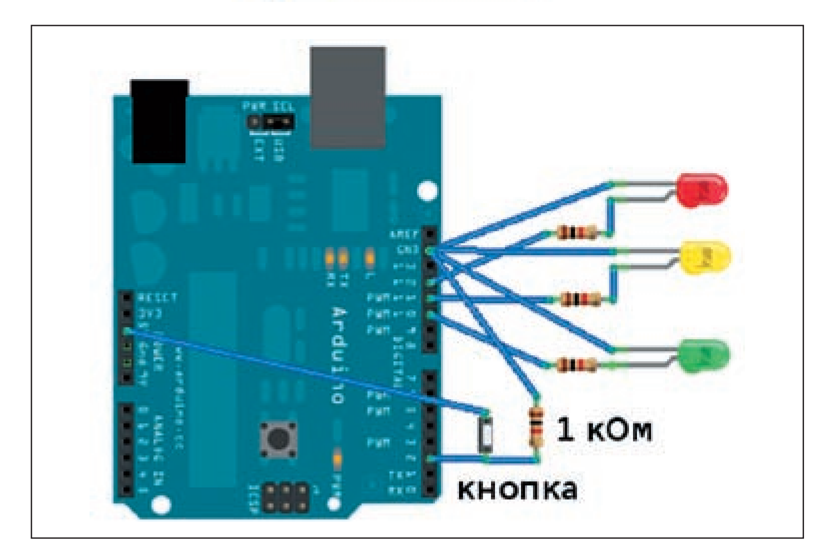
Попробуем смоделировать двухрежимный светофор: изначально он будет работать как обычный (красный/желтый/зеленый), а по нажатию кнопки – переходить в режим мигающего желтого и обратно.

Соберите схему, как показано на рисунке (резисторы нужны для устранения дребезга контактов и ограничения тока на светодиодах). После этого создайте нижеследующий программный код, взяв за основу предыдущий проект. Помните, что персонаж *Catenary* жизненно необходим, и удалять его нельзя.

Разберем алгоритм, который мы применили при создании модели. После инициализации задаются режимы (**^pinMode**) контактов: мы будем использовать цифровые контакты 10, 11 и 12 для управления светодиодами (зеленым, желтым и красным, соответственно) и цифровой контакт под номером 2 как вход кнопки.

```
когда щелкнут по [флаг]
передать ^arduinoPort 0
ждать 0.5 секунд
передать ^reset
ждать 1 секунд
передать ^pinMode 12 output
передать ^pinMode 11 output
передать ^pinMode 10 output
передать ^pinMode 2 input
```

› **Принципиальная схема нашего светофора.**



Считывание данных с кнопки происходит через сенсор **Pin2**. Если его нет в выпадающем списке сенсоров (синяя вкладка Сенсоры), запустите только что набранный код. Как только сенсор фиксирует нажатие, мы изменяем состояние **state** на противоположное.

```
когда щелкнут по [флаг]
всегда
  если Pin2 значение сенсора = 1
    если state = 1
      поставить state в 0
    или
      поставить state в 1
```

После старта мы входим в бесконечный цикл, в котором, в зависимости от значения переменной **state**, реализуем или дневной режим светофора (**state = 0**) или ночной («мигалка» – **state = 1**).

```
когда щелкнут по [флаг]
поставить state в 0
поставить red в 0
поставить yellow в 0
поставить green в 0
всегда
  если state = 0
    поставить yellow в 0
    поставить green в 1
    ждать 2 секунд
    поставить green в 0
    поставить yellow в 1
    ждать 2 секунд
    поставить yellow в 0
    поставить red в 1
    ждать 2 секунд
    поставить red в 0
    поставить yellow в 1
    ждать 2 секунд
```

```
когда щелкнут по [флаг]
всегда
  если state = 1
    поставить green в 0
    поставить red в 0
    поставить yellow в 1
    ждать 0.5 секунд
    поставить yellow в 0
    ждать 0.5 секунд
```



Три переменных (**green**, **red** и **yellow**) определяют, включен ли светодиод соответствующего цвета. Три блока, в зависимости от значения переменных, передают на плату соответствующие уровни для цифровых контактов (**high** и **low**).

Вы можете красиво оформить данную модель, сделав реальный светофор, а также доработать программу, обеспечив правильное переключение света – то есть добавив возможность одновременного включения красного и желтого перед зеленым и наоборот. Также можно поэкспериментировать и с задержками сигналов светофора.

Плата для Scratch

Это все, конечно, интересно, но сложновато для школьников 5–6 класса. А можно ли придумать что-то подобное и для них? Да, если потрудиться и собрать на основе Arduino так называемый ScratchBoard. В оригинале это плата PicoBoard, разработанная специально для использования со *Scratch* и имеющая в нем встроенную поддержку со стороны блоков программирования (http://info.scratch.mit.edu/Sensor_Boards). Проблема в том, что ScratchBoard нужно заказывать из-за рубежа (хотя цена его – не более $50); но с помощью Arduino мы сделаем подобную плату сами. По адресу http://scratch.mit.edu/forums/viewtopic.php?id=28188 можно найти различные варианты реализации и советы. Воспользуемся одним из них: http://www.yengawa.com/scratch_arduino.

> «Для школьников 5–6 класса можно собрать ScratchBoard.»

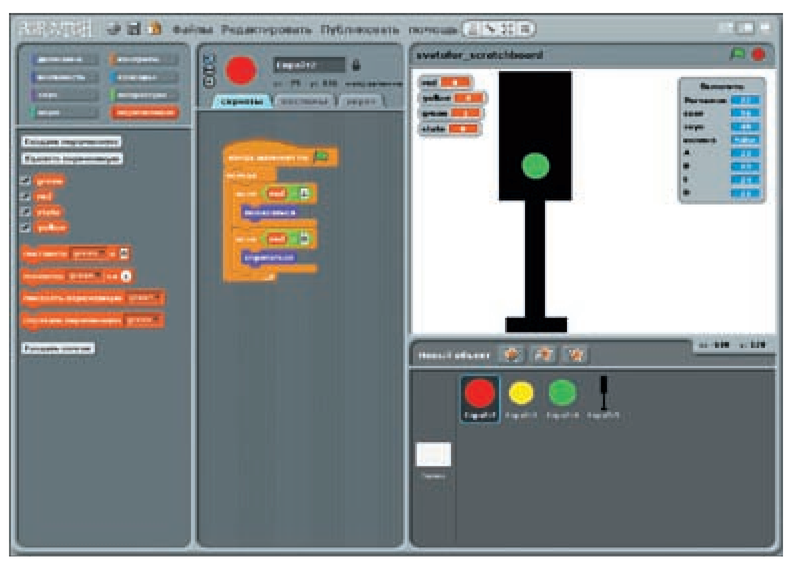
Для начала, реализуем не все функции (это тема для отдельной статьи – напишите нам на letters@linuxformat.ru, если такой материал представляет интерес). Мы воспользуемся кнопкой от предыдущего проекта. Позже можно будет добавить датчики освещения, звука, ползунковый резистор и четыре резистивных входа.

Собрав все вместе и подключив к плате, нужно запустить *Arduino IDE* и залить в устройство прошивку эмулятора, которую можно найти по адресу http://www.yengawa.com/sites/default/files/uploads/ScratchBoard.pde или на **LXF**DVD.

Далее, запустите *Scratch* и, щелкнув правой кнопкой мыши по блоку Значение сенсора в синей вкладке Сенсоры, выберите Показать данные ScratchBoard. Затем щелкните правой кнопкой по блоку, возникшему в окне отображения, и выберите порт, к которому подключена ваша плата. Теперь можете проверить, изменяются ли значения сенсоров при взаимодействии с ними. Если все работает, то можно приступать к программированию (блок можно просто скрыть). У вас может возникнуть проблема с доступом к порту, поэтому если после подключения платы ничего не поменялось, дайте в терминале от имени root команду

```
chown user /dev/ttyUSB0
```

где **user** – ваш пользователь, а **/dev/ttyUSB0** – порт, к которому подключена плата Arduino.



› **Наш светофор не только умеет мигать в двух режимах, но и позволяет переключать их кнопкой на нашей импровизированной ScrAtchrDuinoBoard.**

Теперь можно протестировать нашу видоизмененную программу модели светофора, но уже с использованием возможностей нашей «ScratchBoard» и без реальных светодиодов. Графический текст программы вы можете найти на **LXF**DVD (файл **ScratchBoard_svetofor.pdf**) или (наряду с другими программами) по адресу http://wiki.edumandriva.ru/wiki/index.php/Arduino.

Мы рассмотрели только самые основные моменты использования связки Scratch–Arduino. «За бортом» остались управление двигателями, использование сенсоров, снятие показаний с датчиков и другие не менее интересные вещи. Напишите нам или оставьте сообщение на форуме forum.linuxformat.ru, если данная тема вам интересна и ее стоит развивать. LXF

В июльском номере

Двойная загрузка — это просто!

Linux и Windows вполне могут сосуществовать в мире и согласии – мы покажем, как этого добиться.

Фильтры содержимого

Познакомьтесь с лучшими приложениями, позволяющими защитить вашу семью или офис от нежелательного контента.

Резервная копия жизни

Хотите сохранять все изменения, которые вы когда-либо делали в каком-либо файле? Ну так реализуйте это – по нашему учебнику.

Содержание последующих выпусков может меняться без уведомления.



Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 5000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор
Валентин Синицын info@linuxformat.ru
Литературный редактор
Елена Толстякова
Переводчики
Илья Аввакумов, Александр Бикмеев, Юлия Дронова, Александр Казанцев, Ольга Кокорева, Светлана Кривошеина, Алексей Опарин, Валентин Развозжаев, Татьяна Цыганова
Редактор диска
Александр Кузьменков
Верстка, допечатная подготовка
Сергей Рогожников
Технический директор
Денис Филиппов
Генеральный директор
Павел Фролов
Учредители
Частные лица
Издатели
Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Взлет»
196644, Санкт-Петербург, Колпинский район, пос. Саперный
Тел.: (812) 462-85-85
Заказ 3187

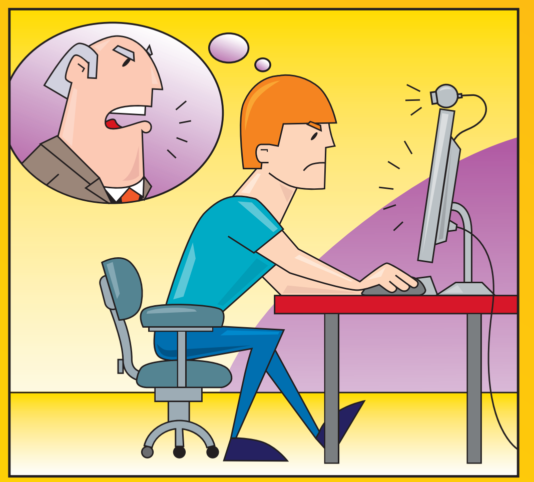
РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

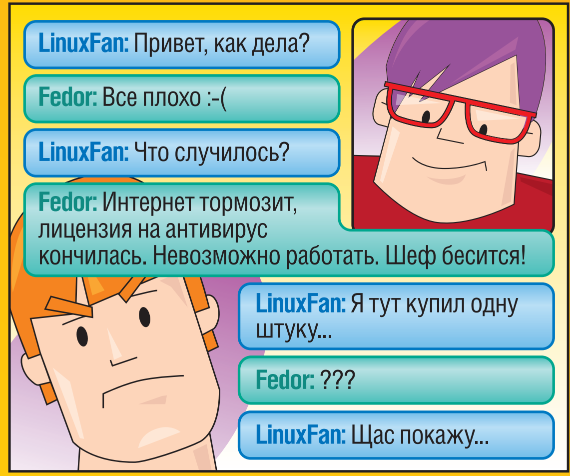
Редактор Пол Хадсон [Paul Hudson] paul.hudson@futurenet.com
Редактор обзоров Грэм Моррисон [Graham Morrison] graham.morrison@futurenet.com
Редактор диска Майк Сондерс [Mike Saunders] mike.saunders@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com
Литературный редактор Эндрю Грегори [Andrew Gregory] agregory@futurenet.com
Подготовка материалов Джоно Бэкон [Jono Bacon], Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Энди Ченнел [Andy Channelle], Алекс Кокс [Alex Cox], Джульетта Кемп [Juliet Kemp], Бен Мартин [Ben Martin], Мартин Мередит [Martin Meredith], Боб Мосс [Bob Moss], Шашанк Шарма [Shashank Sharma], Клаудио Танчони [Claudio Tancioni] Ник Вейч [Nick Veitch], Коэн Верплоесем [Koen Vervloesem], Леонид Алифанов, Евгений Балдин, Андрей Боровский, Александр Казанцев, Евгений Крестников, Александр Кузьменков, Александр Маджугин, Дмитрий Михирев, Григорий Рудницкий, Алексей Федорчук, Вячеслав Ястребцев
Художественные ассистенты Стейси Блек [Stacey Black], Ник Кокс [Nick Cox]
Иллюстрации Крис Хедли [Cris Hedley], Элли Уолтон [Elly Walton], Крис Винн [Chris Winn], iStockPhoto

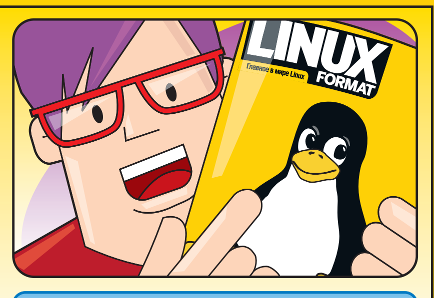
КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel +44 01225 442244 Email: linuxformat@futurenet.com
РОССИЯ:
Санкт-Петербург (редакция):
Лиговский пр., 50, корп. 15
Тел.: +7 (812) 309-06-86
Представительство в Москве:
Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)
Тел/факс: +7 (499) 271-49-54
По вопросам сотрудничества, партнерства, оптовых закупок: partner@linuxcenter.ru



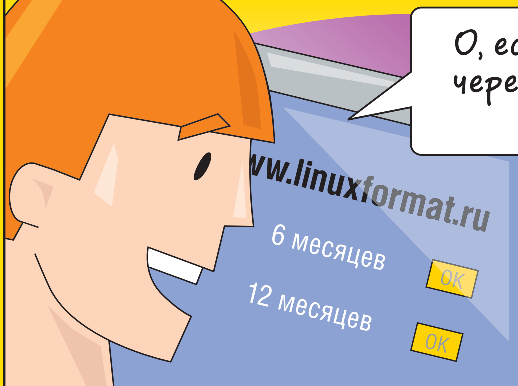


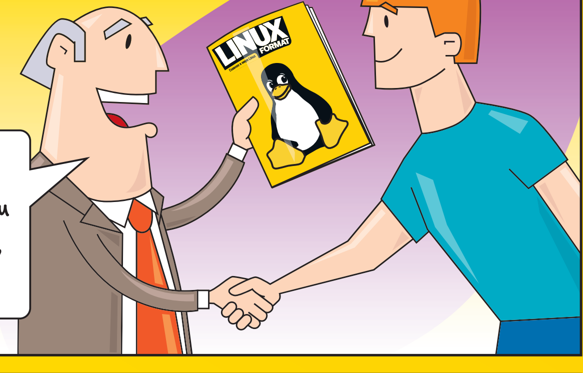


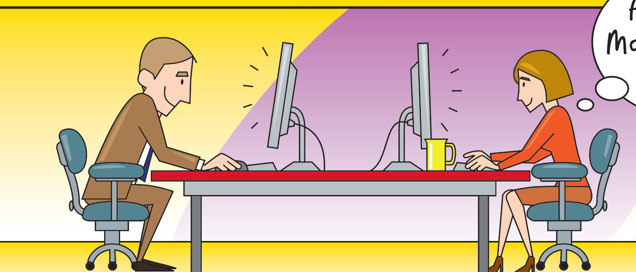


















www.linuxformat.ru

ПОДПИСНЫЕ ИНДЕКСЫ В КАТАЛОГАХ:

Агентство «Роспечать»	20882
«Пресса России»	87974
«Почта России»	16572